CO-CHANNEL INTERFERENCE REDUCTION

Rutgers University

Alvin Garcia and Dr. Richard Mammone

19961021 117

Rome Laboratory
Air Force Materiel Command
Rome, New York

DTIC QUALITY INSPECTED 1

This report has been reviewed by the Rome Laboratory Public Affairs Office (PA) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be releasable to the general public, including foreign nations.

RL-TR-96-147 has been reviewed and is approved for publication.

APPROVED: *John Grieco*

JOHN GRIECO
Project Engineer

FOR THE COMMANDER:

JOSEPH CAMERA
Technical Director
Intelligence & Reconnaissance Directorate

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (Leave Blank)	2. REPORT DATE July 1996	3. REPORT TYPE AND DATES COVERED Final Apr 95 – Apr 96	

4. TITLE AND SUBTITLE

CO-CHANNEL INTERFERENCE REDUCTION

6. AUTHOR(S)

Alvin Garcia and Dr. Richard Mammone

5. FUNDING NUMBERS

C - F30602-95-C-0074
PE - 35885G
PR - 3188
TA - C0
WU - 02

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Rutgers University
Busch Campus, CAIP Center
P. O. Box 1390
Piscataway NJ 08855-1390

8. PERFORMING ORGANIZATION REPORT NUMBER

N/A

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Rome Laboratory/IRAA
32 Hangar Rd
Rome NY 13441-4114

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

RL-TR-96-147

11. SUPPLEMENTARY NOTES

Rome Laboratory Project Engineer: John J. Grieco/IRAA/(315)330-4024

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

Co-channel speaker interference occurs when the voice of one speaker is corrupted by the superposition of another speaker's voice on the same communications channel. The presence of co-channel interference in a communications scenario results in decreased intelligibility (for human listeners) both of the target speaker's speech as well as that of the interfering speaker. While humans can partially compensate for such interference, the performance of automatic speech processing systems, such as speech recognizers and speaker recognition systems, deteriorates drastically in the presence of such interference. This report describes a new methodology for speaker separation in the audio domain. This methodology is based on a frame-by-frame technique which is called the pole-partitioning separation algorithm.

14. SUBJECT TERMS

Co-channel, Interference reduction

15. NUMBER OF PAGES
136

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18
298-102

Contents

   Appendix A - Methods for Co-Channel Speech Separation

List of Tables

1 Introduction

This document has been prepared to satisfy the data item "Scientific and Technical Report (Final)", as specified by Contract Line Item No. CLIN 0002, Data Item No. A003 for the contract PR NO. I-5-4102, "Co-channel Interference Reduction" sponsored by Rome Laboratory.

1.1 Program Goals

The primary goal of this effort was to develop algorithms for performing effective co-channel interference reduction and co-channel speaker separation. The developed algorithms would allow suppression of the interference speaker's voice in the co-channel signal, leaving the voice of the desired target speaker. Furthermore, the algorithms would allow for recovery of the speech of the interference speakers as well.

1.2 Definition of Terms and Abbreviations

As a reference, we provide here a brief definition of various terms and abbreviations used in this report.

AR model auto-regressive signal model.

ARMA model auto-regressive moving-average model.

co-channel signal A monophonic signal consisting of the superposition of two (or more) independent speech signals.

interference speaker In certain cases, the speech of a particular speaker is of interest in the co-channel signal; all other speech signals present are not of interest. In such cases, those speakers whose speech is not of interest are referred to as *interference speakers*. See **target speaker**.

louder/stronger speaker The speaker whose voice signal has the highest energy of all speakers present in a given frame of co-channel speech.

pole-partitioning algorithm The algorithm used to estimate models $S_1(z)$ and $S_2(z)$ from $N_{co}(z)$ and $D_{co}(z)$.

pole-partitioning separation algorithm The separation algorithm consisting of two steps: 1) estimation of $N_{co}(z)$ and $D_{co}(z)$, and 2) estimation of models $S_1(z)$ and $S_2(z)$ from $N_{co}(z)$ and $D_{co}(z)$, accomplished via the pole-partitioning algorithm.

talker/speaker This term is often used interchangeably with "speaker's voice" or "speaker's speech signal".

target speaker In certain cases, the speech of a particular speaker is of interest in the co-channel signal. In such cases, that speaker is referred to as the *target* speaker.

TIR target-to-interference ratio. See **VVR**.

VVR voice-to-voice ratio. The ratio of the energy of one speaker's voice signal to that of the other speaker. Note that we use this term interchangeably with TIR.

SNR signal-to-noise ratio. The ratio of the energy of a signal to the energy of any noise present along with that signal.

1.3   Summary of Results

The heart of our proposed co-channel speaker separation and interference reduction method is the pole-partitioning separation algorithm. The algorithm is described in detail in Section 2.3. However, we summarize it here briefly for the sake of clarity. First, we assume the widely known AR signal model

$$S(z) = \frac{\sigma}{A(z)}. \tag{1}$$

for each of the individual speech signals present in a given frame of co-channel speech. Then for a co-channel signal $s_{co}(n) = s_1(n) + s_2(n)$, the corresponding Z-transform is given by:

$$
\begin{aligned}
S_{co}(z) &= S_1(z) + S_2(z) \\
&= \frac{\sigma_1}{A_1(z)} + \frac{\sigma_2}{A_2(z)}
\end{aligned}
$$

$$= \frac{\sigma_1 A_2(z) + \sigma_2 A_1(z)}{A_1(z) A_2(z)}$$

$$\equiv \frac{N_{co}(z)}{D_{co}(z)} \tag{2}$$

where $S_1(z)$ and $S_2(z)$ are the Z-transforms of the separate speech signals, $s_1(n)$ and $s_2(n)$, respectively.

The pole-partitioning separation algorithm consists of two major processing stages:

1. Estimation of the polynomials $N_{co}(z)$ and $D_{co}(z)$ of the ARMA model $S_{co}(z) = \frac{N_{co}(z)}{D_{co}(z)}$ as defined in Eq. 2.

2. Estimation of the polynomials $A_1(z)$ and $A_2(z)$ and gain factors σ_1 and σ_2 of the AR models $S_1(z) = \frac{\sigma_1}{A_1(z)}$ and $S_2(z) = \frac{\sigma_2}{A_2(z)}$ from the polynomials $N_{co}(z)$ and $D_{co}(z)$ obtained in Step 1. This step is accomplished via the pole-partitioning algorithm.

When the polynomials $N_{co}(z)$ and $D_{co}(z)$ are known a priori, as given by Eq. 2, the pole-partitioning separation algorithm is capable of successfully performing Step 2, i.e. accurately estimating the two AR models $S_1(z)$ and $S_2(z)$ from $N_{co}(z)$ and $D_{co}(z)$. However, in practice, these polynomials $N_{co}(z)$ and $D_{co}(z)$ are not known, and must be estimated directly from the frame of co-channel speech, using ARMA parameter estimation techniques. Unfortunately, while many of the existing ARMA parameter estimation methods are adequate for ARMA *modeling* of data, none of them are sufficiently accurate for ARMA parameter *estimation*, except in the case of noise-free signals generated by a true ARMA filter excited by gaussian noise. When ARMA estimation techniques are used in Step 1 so as to generate estimates $\hat{N}_{co}(z)$ and $\hat{D}_{co}(z)$ of $N_{co}(z)$ and $H_{co}(z)$, respectively, the pole-partitioning separation algorithm fails to successfully perform Step 2. That is to say, when the estimates $\hat{N}_{co}(z)$ and $\hat{D}_{co}(z)$ generated in Step 1 of the pole-partitioning separation algorithm are fed to Step 2, the resulting estimates $\hat{S}_1(z)$ and $\hat{S}_2(z)$ do not, in general, "match" the actual models $S_1(z)$ and $S_2(z)$. The degree to which the estimates "match" the actual models is considered in the spectral magnitude sense, i.e. with $S_1(z)$ replaced by $|S_1(z)|_{z=e^{j\omega}}$, and similarly for $S_2(z)$, $\hat{S}_1(z)$, and $\hat{S}_2(z)$.

In short, the pole-partitioning separation algorithm is capable of performing frame-wise separation when $N_{co}(z)$ and $D_{co}(z)$ are known or can be estimated accurately,

7

but fails otherwise. In that in a practical system, these polynomials are not known a priori, and in so far as these polynomials cannot be accurately estimated with any known existing ARMA parameter estimation techniques, the pole-partitioning separation algorithm fails to accomplish the task of blind separation. As such, we have chosen not to run the tests described in the **Test Plan document** previously submitted, as the results would be inconclusive. In its place, we have run an alternative set of experiments. These are described in detail in Section 4.

1.4   Report Outline

This "Scientific and Technical Report (Final)", hereafter referred to as the **Final Report**, documents the work completed under this contract, contract PR NO. I-5-4102, "Co-channel Interference Reduction". Section 2 provides a technical description of our work and the algorithms developed, particularly the pole-partitioning separation algorithm. Also included in this section is a brief summary of an extensive review of previous research efforts on the co-channel speaker separation task. The complete review, consisting of the Master's Thesis of one of the Research Assistants employed under this contract, is provided in Appendix A. Section 3 describes the methods used to evaluate the quality of the speech recovered by our co-channel speaker separation algorithms. Section 4 describes the results of our experiments. Finally, Section 5 provides a discussion of the results and the conclusions drawn. Additionally, suggestions for future work are provided.

2 Co-channel Speaker Separation System Description

2.1 Introduction

Co-channel speaker interference occurs when the voice of one speaker is corrupted by the superposition of another speaker's voice on the same communications channel. The presence of co-channel interference in a communications scenario results in decreased intelligibility (for human listeners) of both the target speaker's speech as well as that of the interfering speaker. While humans can partially compensate for such interference, the performance of automatic speech processing systems, such as speech recognizers and speaker recognition systems, deteriorates drastically in the presence of such interference. As described in the Statement of Work, the goal of a co-channel interference reduction system is to suppress the voice of the interfering speaker in the co-channel signal, so as to increase the intelligibility of the target speaker. In a co-channel speaker separation system, in addition to the voice of the target speaker, the voice of the interfering speaker should be recovered as well.

2.2 Previous Work

In this section we provide a summary of an extensive review of previous research efforts directed at the task of co-channel interference reduction and speaker separation. The complete review, consisting of the Master's Thesis of one of the Research Assistants employed under this contract, is provided in Appendix A.

2.2.1 Stages of a Co-channel Interference Reduction System

In most every system for co-channel interference reduction, there are a number of processing stages which need to be implemented or otherwise addressed. These include:

1. Determination of the number of people speaking within each analysis frame

2. Determination of the voicing (i.e. voiced or unvoiced) of each speaker within each analysis frame

3. Actual separation of the two voices within each analysis frame

4. Reassembly of the separated frames into contiguous speech utterances.

Step #1 is necessary so that subsequent processing stages do not attempt to separate the input signal into two speech signals, or attenuate part of the speech signal, when only one speaker is present. Depending on the particular separation scheme utilized, attempts to separate a single speech signal into two speech signals can produce meaningless and confusing results. Similarly, no processing should be attempted in intervals of silence, when neither speaker is talking. Step #2 is necessary in order to ensure that the appropriate type of processing is performed on the co-channel signal. Voiced speech, which exhibits highly structured spectral and time-domain characteristics, should be processed differently than unvoiced speech, whose spectral and time-domain characteristics are largely random. For example, a comb-filter might be used to enhance the intelligibility of a segment of voiced speech. However, the application of a comb-filter to a segment of *unvoiced* speech will produce a signal which sounds like voiced speech; this is clearly a misleading result. Many co-channel interference reduction schemes assume the presence of two *voiced* speech signals. As in the case of step #1, such an assumption can result in meaningless and confusing results when both speakers' speech signals aren't voiced. Step #3 refers to the actual separation of a frame of co-channel speech into estimates of each speaker's speech; in the case of a co-channel interference reduction system (as opposed to a co-channel speech separation system), this step entails removing the estimate of the interfering speaker's speech from the co-channel signal so as to leave only the voice of the desired target speaker. Step #4 is necessitated by the fact that step #3 generates estimates of each speaker's speech signal, but it doesn't "know" which signal belongs to which speaker. The frames of separated speech produced by step #3 must be reassembled in such a way as to maintain continuity of speaker identity across frame boundaries so as to properly reconstruct each of the constituent speech utterances.

2.2.2 Separation/Suppression Methods

Most of the past research on co-channel interference reduction has focused on processing step #3 described in the above section. The previous research can be classified

along the following dichotomy: **pitch-based** separation methods and **non-pitch-based** separation methods. In the pitch-based separation methods, separation usually proceeds as follows:

1. Estimate pitch of each speaker's voice

2. Use these estimates to enhance the target speaker's voice (such as by employing a comb-filter tuned to his/her pitch) and/or suppress the interfering speaker's voice (such as by using a multi-tooth notch-filter)

In non-pitch-based methods, separation is accomplished without explicit use of estimates of the constituent pitches. Almost all of the studies reviewed employ separation methods which fall into the former category.

2.2.3   Pitch estimation schemes

Most of the multi-pitch estimation algorithms proposed in the surveyed literature can be grossly classified as either **iterative** or **non-iterative**. In non-iterative algorithms, the pitch of both speakers is estimated simultaneously, in a one-shot fashion. In iterative schemes, the following sequence of steps are usually observed:

1. Generate a single pitch estimate

2. Use this pitch estimate to suppress (as with a multi-tooth notch-filter) the voice of one speaker from the frame of co-channel speech

3. Generate a second pitch estimate from the residual signal left by step #2; presumably this is the pitch of the other speaker's voice

4. (optional) Use this second pitch estimate to suppress the voice of this second speaker from the frame of co-channel speech; return to step #1

In many cases, standard single-pitch estimation schemes were adapted to be multi-pitch estimators by utilizing this sequence of steps.

Most of the pitch estimators employed in the previous studies can also be broadly categorized into one of the following groups:

- auditory-model-based pitch estimation

- Maximum-Likelihood pitch estimation

- ACF-based (autocorrelation function) pitch estimation

- frequency domain peak-picking pitch estimation

Detailed descriptions and a comparative experimental evaluation of these types of pitch estimators can be found in the Appendix.

2.2.4   Conclusions and Future Work

Upon completion of the literature search on previous research and the subsequent evaluation of these studies, we have been led to believe that pitch-based approaches for accomplishing separation of co-channel speech have been exhausted. While such methods offer some degree of separation under certain conditions, their reliance upon the assumption of voiced-on-voiced speech ultimately limits their application to realistic conditions, where speech consists of both voiced and unvoiced segments.

2.3   Approach

2.3.1   Problem Formulation

Often times in speech processing, the sampled speech signal $s(n)$ is modeled as the output of a Pth-order auto-regressive (AR), or all-pole, filter over the duration of a short analysis frame. In the Z-domain, this can be expressed as:

$$S(z) = \frac{E(z)}{A(z)} \tag{3}$$

where $A(z)$ and $E(z)$ are polynomials in z^{-1}. The polynomial $A(z)$ (actually $1/A(z)$) models the frequency characteristics of the vocal and nasal tracts, while $E(z)$ models the frequency characteristics of the excitation signal. In the case of a *voiced* sound, $E(z)$ corresponds to a periodic excitation signal, whereas in the case of an *unvoiced* sound, $E(z)$ corresponds to a white noise signal. In a variety of speech processing

scenarios (some speech recognition systems and speaker identification systems, for instance), only the general spectral shape or spectral envelope of the speech signal, as reflected by $A(z)$, is of interest. In such cases, it is sufficient to model $E(z)$ by a white noise signal with power σ, and the resulting simplified model is given by:

$$S(z) = \frac{\sigma}{A(z)}. \tag{4}$$

Using this model, a co-channel signal $s_{co}(n)$ can be modeled as the superposition of two signals, $s_1(n)$ and $s_2(n)$:

$$s_{co}(n) = s_1(n) + s_2(n). \tag{5}$$

By the linearity of the Z-transform, $S_{co}(z)$, the Z-transform of $s_{co}(n)$, is given by the sum of $S_1(z)$ and $S_2(z)$, the Z-transforms of $s_1(n)$ and $s_2(n)$, respectively. Thus, if $S_1(z)$ and $S_2(z)$ are given by:

$$S_1(z) = \frac{\sigma_1}{A_1(z)} \tag{6}$$

and

$$S_2(z) = \frac{\sigma_2}{A_2(z)}, \tag{7}$$

where $A_1(z)$ and $A_2(z)$ are Pth-order polynomials in z^{-1}, then we have:

$$
\begin{aligned}
S_{co}(z) &= S_1(z) + S_2(z) \\
&= \frac{\sigma_1}{A_1(z)} + \frac{\sigma_2}{A_2(z)} \\
&= \frac{\sigma_1 A_2(z) + \sigma_2 A_1(z)}{A_1(z) A_2(z)} \tag{8} \\
&\equiv \frac{N_{co}(z)}{D_{co}(z)} \tag{9}
\end{aligned}
$$

"Separation" of the two speech signals in a frame of co-channel speech can be accomplished by estimation of the polynomials $A_1(z)$ and $A_2(z)$ and gain factors σ_1 and σ_2, and then resynthesis of the signals $s_1(n)$ and $s_2(n)$ via Eq. (6) and Eq. (7). Thus, to accomplish practical speaker separation, it is necessary to be able to estimate these parameters directly from the co-channel signal.

13

2.3.2 Algorithm Description

The method we have been developing for estimation of the polynomials $A_1(z)$ and $A_2(z)$ and gain factors σ_1 and σ_2 from a frame of co-channel speech consists of two processing stages:

1. Estimation of the polynomials $N_{co}(z)$ and $D_{co}(z)$ (as defined in Eq. 9).

2. Determination of $A_1(z)$ and $A_2(z)$ (and σ_1 and σ_2) from $N_{co}(z)$ and $D_{co}(z)$.

The second step is accomplished via the **pole-partitioning** algorithm, and collectively, we call these two steps the **pole-partitioning separation algorithm**.

Estimation of $N(z)$ and $D(z)$ The numerator $N_{co}(z)$ in Eq. (8) is a Pth degree polynomial, while the denominator $D_{co}(z)$ is a polynomial of degree $2P$. The task of estimating the numerator $N(z)$ and denominator $D(z)$ from a data set, such that the filter $H(z) = N(z)/D(z)$ "best" models the data set (assuming that the filter was driven by a white noise sequence), is commonly referred to as ARMA (Auto Regressive Moving Average) system identification. In general, the polynomials $N(z)$ and $D(z)$ are independent of each other. However, in this case, as shown by Eq. (8) and Eq. (9), we have $N(z) = N_{co} \equiv \sigma_1 A_2(z) + \sigma_2 A_1(z)$ and $D(z) = D_{co}(z) \equiv A_1(z)A_2(z)$. Obviously, $N(z)$ and $D(z)$ are not independent here. Consequently, the constraint imposed by these relationships must not be violated when generating estimates of $N_{co}(z)$ and $D_{co}(z)$. There are a number of standard techniques available for performing ARMA system identification. These include Prony's method, Durbin's method, Shanks' method, and the iterative prefiltering method of Steiglitz and McBride (See [10] and [3]). There is also a relatively new iterative Prony method [11]. None of these methods, however, allow the incorporation of the constraints imposed by the particular structure of $N_{co}(z)$ and $D_{co}(z)$ which is present in this situation. Furthermore, a problem common to these methods is that accurate estimation of $N(z)$ and $D(z)$, especially $N(z)$, is rarely achieved unless the data sequence corresponds exactly to a sequence generated by an ARMA filter; most of these methods are better suited for data *modeling* rather than for system *identification*. Nevertheless, these algorithms represent the standard methods that are available for ARMA parameter estimation.

14

Determination of $A_1(z)$ and $A_2(z)$ (and σ_1 and σ_2) from $N_{co}(z)$ and $D_{co}(z)$: Pole-Partitioning Determination of $A_1(z)$ and $A_2(z)$ (and σ_1 and σ_2) from $N_{co}(z)$ and $D_{co}(z)$ is a non-trivial task. This is due to the fact that the relationship between the coefficients of $N_{co}(z)$ and $D_{co}(z)$ and the coefficients of $A_1(z)$ and $A_2(z)$ is non-linear. This is most readily demonstrated with a simple example. Consider the case where $P = 3$,

$$A_1(z) = a_0 + a_1 z^{-1} + a_2 z^{-2} + a_3 z^{-3} \tag{10}$$

and

$$A_2(z) = b_0 + b_1 z^{-1} + b_2 z^{-2} + b_3 z^{-3}, \tag{11}$$

and $N_{co}(z)$ and $D_{co}(z)$ have been estimated already. Then the product of $A_1(z)$ and $A_2(z)$, which corresponds to $D_{co}(z)$, is given by:

$$
\begin{aligned}
A_1(z)A_2(z) = \ & (a_0 b_0) \\
+ \ & (a_0 b_1 + a_1 b_0)z^{-1} \\
+ \ & (a_0 b_2 + a_1 b_1 + a_2 b_0)z^{-2} \\
+ \ & (a_0 b_3 + a_1 b_2 + a_2 b_1 + a_3 b_0)z^{-3} \\
+ \ & (a_0 b_4 + a_1 b_3 + a_2 b_2 + a_3 b_1 + a_4 b_0)z^{-4} \\
+ \ & (a_0 b_5 + a_1 b_4 + a_2 b_3 + a_3 b_2 + a_4 b_1 + a_5 b_0)z^{-5} \\
+ \ & (a_0 b_6 + a_1 b_5 + a_2 b_4 + a_3 b_3 + a_4 b_2 + a_5 b_1 + a_6 b_0)z^{-6}. \tag{12}
\end{aligned}
$$

Likewise, the weighted sum of $A_1(z)$ and $A_2(z)$, corresponding to $N_{co}(z)$, is given by:

$$
\begin{aligned}
\sigma_1 A_2(z) + \sigma_2 A_1(z) = \ & \sigma_1(b_0 + b_1 z^{-1} + b_2 z^{-2} + b_3 z^{-3}) \\
& + \sigma_2(a_0 + a_1 z^{-1} + a_2 z^{-2} + a_3 z^{-3}) \\
= \ & (\sigma_1 b_0 + \sigma_2 a_0) + (\sigma_1 b_1 + \sigma_2 a_1)z^{-1} \\
& + (\sigma_1 b_2 + \sigma_2 a_2)z^{-2} + (\sigma_1 b_3 + \sigma_2 a_3)z^{-3} \tag{13}
\end{aligned}
$$

Thus, formulating the equation:

$$\frac{N(z)}{D(z)} = \frac{\sigma_1 A_2(z) + \sigma_2 A_1(z)}{A_1(z)A_2(z)} \tag{14}$$

with $N(z)$ and $D(z)$ known, and equating coefficients of like powers of z^{-1} of the numerators on both sides of the equation, and similarly for the denominators, we are left with a set of $(2P-1)+(P+1) = 3P$ nonlinear equations in $2P+2$ unknowns, namely the coefficients of $A_1(z)$ and $A_2(z)$ and the gain factors σ_1 and σ_2. An analytical solution to this problem is highly intractable, if not outright unobtainable. Solution

by numerical methods may be attempted. However, as the equations are highly non-linear, the associated error function is not likely to be convex, and thus numerical methods may converge to local minima solutions rather than the global minimum, depending on the initial conditions of the numerical solution method. Obviously, such sub-optimal solutions are unacceptable if accurate determination of $A_1(z)$, $A_2(z)$, σ_1, and σ_2 are to be obtained. To circumvent this situation, we have developed a new method for solving for these parameters given $N(z)$ and $D(z)$: the pole-partitioning algorithm.

Pole-Partitioning Algorithm Inspection of Eq. (8) shows that the poles of $S_{co}(z)$ (i.e. the roots of $D_{co}(z)$) consist of the union of the set of roots of $A_1(z)$ and the set of roots of $A_2(z)$. This follows because a given Pth-order polynomial $A(z)$ can be expressed as the product of P first-order terms. Mathematically, this is given by:

$$A(z) = \sum_{i=0}^{P} a_i z^{-i} = \prod_{i=1}^{P}(1 - z_i z^{-1}) \tag{15}$$

where the $\{z_i\}_{i=1...P}$ are the P roots of $A(z)$. Thus, the denominator $D_{co}(z) = A_1(z)A_2(z)$ is:

$$
\begin{aligned}
A_1(z)A_2(z) &= \prod_{i=1}^{P}(1 - z_{1i} z^{-1}) \prod_{j=1}^{P}(1 - z_{2j} z^{-1}) \\
&= \prod_{k=1}^{2P}(1 - z_k z^{-1}) \tag{16}
\end{aligned}
$$

where $\{z_{1i}\}_{i=1...P}$ are the P roots of $A_1(z)$, $\{z_{2j}\}_{j=1...P}$ are the P roots of $A_2(z)$, and $\{z_k\}_{k=1...2P}$ are the roots of $A_1(z)$ and roots of $A_2(z)$ taken collectively. Thus, if these $2P$ roots can be estimated, then the denominator $D_{co}(z)$ of $S_{co}(z)$ can be determined trivially, by the product indicated in Eq. (15). If then the roots of the numerator $N_{co}(z)$ can be estimated, or equivalently, if the polynomial $N_{co}(z)$ itself can be estimated directly, then what remains is the determination of $A_1(z)$ and $A_2(z)$ (and σ_1 and σ_2) from $N_{co}(z)$ and $D_{co}(z)$.

The pole-partitioning algorithm works as follows: Given the $2P$ poles of $D_{co}(z)$, there are a total of $\binom{2P}{P}$ ways of partitioning them into two groups of P poles each. For a given partitioning, one of these two groups of P poles corresponds to an

16

estimate $\hat{A}_1(z)$ of $A_1(z)$, and the other group corresponds to an estimate $\hat{A}_2(z)$ of $A_2(z)$. The polynomials $A_1(z)$ and $A_2(z)$ and gain factors σ_1 and σ_2 are estimated by choosing those estimates $\hat{A}_1(z)$ and $\hat{A}_2(z)$ (corresponding to a particular partitioning) and choice of σ_1 and σ_2 which minimize the "difference" between

$$\frac{N_{co}(z)}{D_{co}(z)},\tag{17}$$

as obtained in the previous processing stage (Section 2.3.2), and

$$\frac{\hat{N}_{co}(z)}{\hat{D}_{co}(z)} \equiv \frac{\hat{\sigma}_1 \hat{A}_2(z) + \hat{\sigma}_2 \hat{A}_1(z)}{\hat{A}_1(z)\hat{A}_2(z)}\tag{18}$$

over all $\begin{pmatrix} 2P \\ P \end{pmatrix}$ partitionings and over the range of acceptable values for σ_1 and σ_2.

There are a number of different ways to measure the "difference" described above. Note that irrespective of the partitioning chosen, or values of $\hat{\sigma}_1$ and $\hat{\sigma}_2$, the denominator $\hat{A}_1(z)\hat{A}_2(z)$ does not change. Thus, when $N_{co}(z)$ can be estimated accurately, a simple metric to use is simply the Euclidean norm between $N_{co}(z)$ and $\hat{N}_{co}(z)$:

$$||N_{co}(z) - \hat{N}_{co}(z)||_2.\tag{19}$$

However, as indicated in Section 2.3.2, accurate estimation of $N_{co}(z)$ is rarely accomplished via the existing ARMA estimation methods. Therefore, the use of such a direct metric, as described by Eq. 19, can be problematic; if $N_{co}(z)$ is not a reliable estimate, then there is no point is using it in the metric minimization procedure of the pole-partitioning algorithm. However, even though an accurate *parametric* estimate of $N_{co}(z)$ may be difficult, if not impossible, to obtain, it is still possible to measure the "difference" between the models of Eq. 17 and Eq. 18 by measuring some kind of spectral "difference" between the magnitude spectra represented by each of these equations. One such method is the Itakura spectral distortion metric [2] used for measuring spectral differences between a "reference" signal model $\frac{\sigma_R}{A_R(z)}$ and a "test" signal model $\frac{\sigma_T}{A_T(z)}$. It is defined as:

$$d(A_T(z), A_R(z)) = \log\left(\int_{-\pi}^{\pi} \left|\frac{A_R(\omega)}{A_T(\omega)}\right|^2 \frac{d\omega}{2\pi}\right).\tag{20}$$

Eq. (20) represents a measure of the flatness of the spectral ratio $\left|\frac{A_R(\omega)}{A_T(\omega)}\right|$; if the magnitude of the two spectra are equal, then the integral will be equal to unity, and

the resulting value of the metric will be $\log(1) = 0$. There are a few points to note about this metric. First of all, it is not a metric in the traditional sense in that it is not symmetric. This is easily remedied by defining a new metric $\hat{d}(A_T(z), A_R(z))$ as:

$$\hat{d}(A_T(z), A_R(z)) = \hat{d}(A_R(z), A_T(z)) = d(A_R(z), A_T(z)) + d(A_T(z), A_R(z)) \qquad (21)$$

where $d(A_T(z), A_R(z))$ is as defined in Eq. (20). A second consideration in using Eq. (20) or Eq. (21) is that they apply only to AR models. ARMA models can be accommodated by making the following modification to the metric defined by Eq. (20):

$$\tilde{d}(S_T(z), S_R(z)) = \tilde{d}(S_R(z), S_T(z)) = \log\left(\int_{-\pi}^{\pi} \left|\frac{S_T(\omega)}{S_R(\omega)}\right|^2 \frac{d\omega}{2\pi}\right) + \log\left(\int_{-\pi}^{\pi} \left|\frac{S_R(\omega)}{S_T(\omega)}\right|^2 \frac{d\omega}{2\pi}\right). \qquad (22)$$

Here, $S_R(\omega) = S_R(z)|_{z=e^{j\omega}}$, where $S_R(z)$ represents the polynomial ratio $\frac{N_R(z)}{D_R(z)}$ corresponding to the reference signal. Similarly, $S_T(\omega) = S_T(z)|_{z=e^{j\omega}}$, where $S_T(z)$ represents the polynomial ratio $\frac{N_T(z)}{D_T(z)}$ corresponding to the test signal to be compared with the reference signal. In the case of the pole-partitioning algorithm, we have $S_R(z) = S_{co}(z)$, where $S_{co}(z) = \frac{N_{co}(z)}{D_{co}(z)}$, as determined in the previous processing stage detailed in Section 2.3.2. $S_T(z)$ corresponds to the polynomial ratio

$$\frac{\hat{N}_{co}(z)}{\hat{D}_{co}(z)} \qquad (23)$$

as defined in Eq. (18), which represents an estimate of $\frac{N_{co}(z)}{D_{co}(z)}$ corresponding to a particular pole-partitioning and choice of σ_1 and σ_2.

As stated above, sometimes it is impossible to get accurate parametric estimates of $N_{co}(z)$. In such cases, the metric described by Eq. (22) can be used by using *non-parametric* estimates of the magnitude spectra $S_R(\omega) = S_{co}(\omega)$ in Eq. (22). For instance, the discrete-time Fourier Transform (DTFT) can be used to obtain estimates of these quantities. Estimates may also be obtained using cepstrally-smoothed versions of the DTFT [5].

Summary of pole-partitioning separation algorithm The pole-partitioning separation algorithm can be summarized as follows:

1. Generate estimates of $N_{co}(z)$ and $D_{co}(z)$ using one of the standard ARMA estimation techniques listed in Section 2.3.2.

2. Estimate $A_1(z)$, $A_2(z)$, σ_1, and σ_2 by finding that partitioning and choice of $\hat{\sigma}_1$ and $\hat{\sigma}_2$ that minimizes the value of the quantity $\tilde{d}(S_{co}(z), \hat{S}_{co}(z))$, where $\tilde{d}(,)$ is as defined in Eq. (22), $S_{co}(z) = \frac{N_{co}(z)}{D_{co}(z)}$, and $\hat{S}_{co}(z)$ is as defined by Eq. (18).

A number of practical points should be emphasized. First of all, as stated previously, accurate estimation of $N_{co}(z)$ is rarely achievable. Thus, in Step 2, a non-parametric estimate (e.g. DTFT) of $S_{co}(z)$ can be used. However, in Step 1, estimation of $D_{co}(z)$ is still necessary, as the roots of this polynomial are used in the pole-partitioning portion of the algorithm.

A second consideration concerns the actual implementation of the minimization procedure used in the pole-partitioning. There are a finite number of different pole-partitionings, namely $\binom{2P}{P}$ combinations. Minimization across this range of partitions can be accomplished by exhaustively going through all combinations. However, the parameters σ_1 and σ_2 are real scalars; they can take on an infinite number of values, namely the range of real numbers. Obviously, it is not possible to exhaustively find a minimum over an infinite range of real numbers. Even if the values of σ_1 and σ_2 are constrained to lie with a finite range, they can still assume an infinite number of real values within this range. To accommodate this problem, first, instead of minimizing over σ_1 and σ_2 (for a given partitioning) independently, we minimize over the *ratio* $\frac{\sigma_1}{\sigma_2}$ and change the metric of Eq. (22) slightly so as to be invariant to the *absolute* magnitudes of σ_1 and σ_2, being sensitive only to the *relative* magnitudes of σ_1 and σ_2. The modified metric is given by:

$$
\begin{aligned}
\breve{d}(S_T(z), S_R(z)) &= \breve{d}(S_R(z), S_T(z)) \\
&= \log\left(\int_{-\pi}^{\pi} \left| \frac{|S_T(\omega)| - \int_{-\pi}^{\pi} |S_T(\omega)| \frac{d\omega}{2\pi}}{|S_R(\omega)| - \int_{-\pi}^{\pi} |S_R(\omega)| \frac{d\omega}{2\pi}} \right|^2 \frac{d\omega}{2\pi} \right) \\
&\quad + \log\left(\int_{-\pi}^{\pi} \left| \frac{|S_R(\omega)| - \int_{-\pi}^{\pi} |S_R(\omega)| \frac{d\omega}{2\pi}}{|S_T(\omega)| - \int_{-\pi}^{\pi} |S_T(\omega)| \frac{d\omega}{2\pi}} \right|^2 \frac{d\omega}{2\pi} \right). \quad (24)
\end{aligned}
$$

The integrals in the numerator and denominators of each term represent the mean value of the magnitude of the corresponding spectrum. Subtracting these terms has the effect of making the metric sensitive only to the relative magnitudes of σ_1 and

σ_2, not their absolute magnitudes. Then we limit the range of allowable values of the ratio $\frac{\sigma_1}{\sigma_2}$ to lie within a finite range corresponding to the range of expected values of this ratio. Finally, this finite range is sampled at ten equally-spaced points in the interval. The choice of ten sampling points is somewhat arbitrary, and represents a compromise between adequate resolution of the ratio $\frac{\sigma_1}{\sigma_2}$ and processing time. Thus, for each of the $\begin{pmatrix} 2P \\ P \end{pmatrix}$ partitionings, the minimum value of the metric of Eq. (24) for that particular partitioning is found by computing the value of this metric at all ten points in the $\frac{\sigma_1}{\sigma_2}$ range. The global minimum value of the distance metric is found by finding the minimum of these $\begin{pmatrix} 2P \\ P \end{pmatrix}$ minima (one minimum per partitioning).

3 Algorithm Performance Evaluation

This section is a restatement of the test plan described in the previously submitted **Test Plan** document, repeated here for convenience.

3.1 Database

The database to be used for testing will consist of recordings taken from the "New England" dialect region of the TIMIT database [6]. The recordings in this database were conducted in a low noise environment. This is important, since in order to study the effect of different SNRs on the separation algorithm, baseline performance must be established using the "clean" (no added noise) signals. Each recording in the TIMIT database is monophonic, and consists of a single sentence of read speech, spoken by one of a number of different speakers, including both women and men. Co-channel signals are synthesized by simply summing two or more of these recordings sample-wise. In these experiments, we will only be considering the case of at most two people speaking simultaneously on a communications channel. In this case, the co-channel signal $s(n)$ would be synthesized from the appropriately normalized single-speaker recordings $\{s_1(n)|n = 1, 2, \ldots, N_1\}$ and $\{s_2(n)|n = 1, 2, \ldots, N_2\}$ as follows:

$$s(n) = s_1(n) + s_2(n), \quad n = 1, 2, \ldots, \min(N_1, N_2).$$

The recordings to be used in testing are taken from the "testing" segment of the dialect region #1 (New England) portion of the TIMIT database. The particular recordings chosen are listed in Table 1. The "SEX", "SPEAKER ID", and "SENTENCE ID" fields are from the identification nomenclature used in the TIMIT database. First these recordings are downsampled to 8 kHz from the original 16 kHz sampling rate. Then, from these ten recordings, five different basic co-channel recordings are synthesized. These are listed in Table 2. For each of these five basic co-channel recordings, a number of different versions are created for testing various parameters, such as VVR and SNR. This is described in detail in the corresponding following sections.

index	SEX	SPEAKER ID	SENTENCE ID
1	m	cpm0	sa1.txt
2	m	dac0	sa2.txt
3	m	dpk0	si1053.txt
4	m	edr0	si1374.txt
5	m	grl0	si1497.txt
6	m	jeb1	si1467.txt
7	m	jwt0	si1291.txt
8	m	kls0	si1437.txt
9	m	klw0	si1571.txt
10	m	mgg0	si1079.txt

Table 1: Sentences from TIMIT database to be used in experiments.

sentence	sentence 1 index #	sentence 2 index #
A	1	2
B	3	4
C	5	6
D	7	8
E	9	10

Table 2: Synthesized co-channel signals and their constituent single-speaker recordings (Index numbers correspond to entries in Table 1).

3.2 Performance Measures

Both subjective and objective criteria may be used to grade the performance of a co-channel speaker separation algorithm. Subjective criteria typically employ the aggregated opinions of several human listeners. Objective criteria include spectral distortion measures, such as those discussed in [1]. In that subjective measures are by their very nature difficult, if not impossible, to replicate, we have decided to use strictly objective metrics for evaluating the quality of the speech recovered by our speaker separation algorithm. In particular, we have chosen a modification of the Itakura-Saito spectral distortion metric [2] with which to measure the quality of the separated speech frames. The Itakura-Saito metric is defined as:

$$d(H_R(z), H_T(z)) = \log \left[\int_{-\pi}^{\pi} \left| \frac{A_R(z)}{A_T(z)} \right| \frac{d\omega}{2\pi} \right] \tag{25}$$

where $A_R(z)$ and $A_T(z)$ are the Pth order polynomials corresponding to the linear predictive coding (LPC) models

$$H_R(z) = \frac{\sigma_R}{A_R(z)} \quad \text{and} \quad H_T(z) = \frac{\sigma_T}{A_T(z)} \tag{26}$$

of the reference and training speech frames, respectively. Such LPC modeling is well known and is discussed in greater detail in Section 2.3.1 However, digressing briefly, we simply restate that for short time frames, a speech signal $s(n)$ may be modeled as the output of an IIR filter driven by an excitation signal $e(n)$. In the Z-domain, this is given by:

$$S(z) = \frac{E(z)}{A(z)} \tag{27}$$

where $E(z)$ is the Z-transform of $e(n)$, $\frac{1}{A(z)}$ is the IIR filter modeling the magnitude frequency response of the vocal tract, nasal tract, and lips, and $S(z)$ is the Z-transform of the resulting speech signal $s(n)$. In many applications only the spectral envelope, as described by $\frac{1}{A(z)}$, is of interest, and $E(z)$ is reduced to a constant σ, which represents the overall gain (power) of the speech signal. The resulting simplified model is given by:

$$S(z) = \frac{\sigma}{A(z)}. \tag{28}$$

Returning now to Eq. 25, we note that this measure is not symmetric, that is

$$d(H_R(z), H_T(z)) \neq d(H_T(z), H_R(z)),$$

23

and as such, it does not fully qualify as a metric, in the strict mathematical sense. This is easily remedied by defining a new metric $\tilde{d}$ as:

$$\tilde{d}(H_R(z), H_T(z)) = d(H_R(z), H_T(z)) + d(H_T(z), H_R(z)) \tag{29}$$

where $d(H_R(z), H_T(z))$ is the Itakura-Saito metric as defined in Eq. 25.

This new metric is used in the following manner for each of co-channel recordings to be used for testing. First, the co-channel recording is segmented into non-overlapping frames of 32 ms each. No window is applied; i.e. a 32 ms rectangular window is used. The same procedure is applied to each of the two constituent single-speaker recordings used in synthesizing the particular co-channel recording, such that the frames are time-aligned with the frames in the co-channel recording. Next, these two single-speaker files are submitted to a processing step which marks each frame as either "silence" or "non-silence". This procedure is described in Section 3.3.3. For each frame of the co-channel recording, the corresponding frame in each of the two constituent single-speaker recordings is checked for "silence"/"non-silence". If both frames of the single-speaker recordings are marked as "non-silence", then that frame of the co-channel recording is considered to contain two speakers' voices. Only those frames of the co-channel recording, which are marked as containing two voices, are submitted to the pole-partitioning separation algorithm; the rest are not processed. For each input frame selected in this way, the pole-partitioning separation algorithm generates the LPC model estimates

$$\hat{H}_1(z) = \frac{\hat{\sigma}_1}{\hat{A}_1(z)} \quad \text{and} \quad \hat{H}_2(z) = \frac{\hat{\sigma}_2}{\hat{A}_2(z)} \tag{30}$$

corresponding to the two speech signals present in the frame. These models are to be compared with the LPC models

$$H_1(z) = \frac{\sigma_1}{A_1(z)} \quad \text{and} \quad H_2(z) = \frac{\sigma_2}{A_2(z)} \tag{31}$$

which are estimated directly from the corresponding frames of the individual single-speaker speech recordings. Note that the subscripts "1" and "2" in Eq. 30 are arbitrary, and that such numbering does not reflect the identity of the speaker, but merely serves to distinguish between the two model estimates. That is to say that in a given frame, $\hat{H}_1(z)$ may correspond to the voice of Speaker #1, and in the next frame, $\hat{H}_1(z)$ may correspond to the voice of Speaker #2. This being the case, we cannot simply compute the distances $\tilde{d}(H_1(z), \hat{H}_1(z))$ and $\tilde{d}(H_2(z), \hat{H}_2(z))$, as $H_1(z)$ and

24

$\hat{H}_1(z)$ may not correspond to the same speaker, and similarly for $H_2(z)$ and $\hat{H}_2(z)$. To circumvent this possible mismatch, the following two quantities are computed:

$$D_{match} = \tilde{d}(H_1(z), \hat{H}_1(z)) + \tilde{d}(H_2(z), \hat{H}_2(z)) \tag{32}$$

and

$$D_{mismatch} = \tilde{d}(H_1(z), \hat{H}_2(z)) + \tilde{d}(H_2(z), \hat{H}_1(z)). \tag{33}$$

The quantity D_{match} corresponds to the case where $H_1(z)$ and $\hat{H}_1(z)$ correspond to the same speaker, and similarly for $H_2(z)$ and $\hat{H}_2(z)$. The quantity $D_{mismatch}$ corresponds to the case where $H_1(z)$ and $\hat{H}_1(z)$ correspond to *opposite* speakers, and similarly for $H_2(z)$ and $\hat{H}_2(z)$; i.e. when $H_1(z)$ and $\hat{H}_2(z)$ correspond to the same speaker and $H_2(z)$ and $\hat{H}_1(z)$ correspond to the same speaker. The final measure of the separation performance for the ith input frame of co-channel speech is taken as the minimum of these two quantities: D_{match} and $D_{mismatch}$:

$$D(i) = \min(D_{match}, D_{mismatch}). \tag{34}$$

This procedure is repeated for all N frames of co-channel speech, and the final measure of performance for the particular co-channel recording is taken as the average value of $\{D(i)|i = 1, 2, \ldots, N\}$:

$$D_{avg} = \frac{1}{N} \sum_{i=1}^{N} D(i). \tag{35}$$

3.3   Testing Parameters

3.3.1   Voice-to-Voice Ratio (VVR)

The VVR is the ratio of the average power of the voice of one speaker, arbitrarily designated as speaker #1, to the average power of the voice of the other speaker, designated as speaker #2. To synthesize a co-channel recording with a VVR of X dB, first each of the two files to be summed are normalized to unit variance. This is accomplished by computing the standard deviation σ of each recording and then dividing each sample of the recording by this quantity. If the recordings to be summed contain substantial amounts of silence, this normalization is computed in such a way that the variance *over the frames of non-silence* is unity. That is, the frames containing silence are not used in the computation of the variance of the recording.

25

This is to ensure that the intervals of silence do not bias the estimate of the variance of the actual speech in the recordings. Detection of frames of silence is discussed in Section 3.3.3. After the two recordings $\{s_1(n)|n = 1, 2, \ldots, N_1\}$ and $\{s_2(n)|n = 1, 2, \ldots, N_2\}$ have been normalized in this way, they are summed as follows:

$$s(n) = \alpha s_1(n) + s_2(n), \quad n = 1, 2, \ldots, \min(N_1, N_2),$$

where $\alpha = 10^{\frac{XdB}{20dB}}$. This produces a co-channel recording $\{s(n)|n = 1, 2, \ldots, \min(N_1, N_2)\}$ with a (speaker #1/speaker #2) VVR of X dB. Note that all computations up to this point are carried out in double-precision floating-point arithmetic. Finally, the co-channel recording is normalized and requantized to 16-bit linear format. This requantization step is necessary because, in practice, the co-channel signal received for processing will typically be sampled and quantized to a fixed precision (e.g. 16-bit linear encoding) prior to processing via DSP methods, even if these DSP operations are performed using floating-point arithmetic. The normalization of the co-channel recording prior to requantization consists of scaling each sample of the co-channel signal such that the maximum sample amplitude in the recording is equal to the maximum signal amplitude allowed by 16-bit linear encoding. If s_{max} represents the maximum signal amplitude in the recording prior to normalization and x_{max} represents the maximum signal amplitude allowed by the encoding scheme (+32767 in the case of 16-bit linear encoding), then the co-channel recording is normalized by multiplying each sample by the quantity $\frac{x_{max}}{s_{max}}$. This ensures that the full dynamic range of the encoding method is utilized. After completion of these preliminary processing steps, the co-channel recording is then ready to be used for testing the separation algorithm.

As discussed in Section 3.1, there are five basic co-channel recordings, each consisting of the sum of two individual recordings. For each of these recordings A–E listed in Table 2, four different versions, each corresponding to a different VVR, are created. These are listed in Table 3. Each of these $5 \times 4 = 20$ recordings will then be submitted to the performance evaluation procedure described in Section 3.2, and for each recording, a corresponding performance score D_{avg}, as described by Eq. 35, is computed. Each D_{avg} score will fill one entry of the table shown in Table 4.

3.3.2 Signal-to-Noise Ratio (SNR)

The SNR of a given recording is the ratio of the average power of the actual signal in the recording, to the average power of any noise present along with the signal. To

version	VVR
i	0 dB
ii	6 dB
iii	12 dB
iv	18 dB

Table 3: Different VVR versions of the co-channel signals.

Sentence	VVR				avg
	0 dB	6 dB	12 dB	18 dB	
A	–	–	–	–	–
B	–	–	–	–	–
C	–	–	–	–	–
D	–	–	–	–	–
E	–	–	–	–	–
avg	–	–	–	–	

Table 4: Table of D_{avg} scores to be obtained by testing under different VVRs.

synthesize a co-channel signal with a SNR of X dB, we follow a procedure analogous to that described in Section 3.3.1 for creating a co-channel signal with a given VVR. First, the "clean" (no added noise) co-channel signal is normalized to unit variance. Again, this is accomplished by computing the standard deviation σ of the recording, this time the co-channel signal, and then dividing each sample of the recording by this quantity. Also, as in the previous section on VVR, silence frames are excluded from the computation of the variance of the recording. After the co-channel signal $\{s(n)|n = 1, 2, \ldots, N\}$ is normalized, white noise $e(n)$ is added as follows:

$$s_{noisy}(n) = \alpha s(n) + e(n), \quad n = 1, 2, \ldots, N.$$

Here, $\{e(n)|n = 1, 2, \ldots, N\}$ are white noise samples from a uniform probability density distribution with zero mean and unit variance, and $\alpha = 10^{\frac{XdB}{20dB}}$. This produces a noisy co-channel recording $\{s_{noisy}(n)|n = 1, 2, \ldots, N\}$ with a SNR of X dB. Finally, the noise-corrupted co-channel signal $s_{noisy}(n)$ is normalized and requantized to 16-bit linear format, in exactly the same manner as described previously in Section 3.3.1.

Note that the SNR and VVR are independent, so theoretically both SNR *and* VVR can be varied simultaneously. The primary purpose of this particular experiment is to investigate the effect of additive noise on the performance of the separation algorithm. Arguably, such additive noise may affect recovery of the louder and the quieter speaker in different ways. However, as a practical consideration to limit the experiment variations to a manageable number, we have decided to arbitrarily fix the VVR at 0 dB, while varying the SNR.

As discussed in Section 3.1, there are five basic co-channel recordings, each consisting of the sum of two individual recordings. For each of these recordings A–E listed in Table 2, four different versions, each corresponding to a different SNR, are created. These are listed in Table 5. Each of these $5 \times 4 = 20$ recordings will then be submitted to the performance evaluation procedure described in Section 3.2, and for each recording, a corresponding performance score D_{avg}, as described by Eq. 35, is computed. Each D_{avg} score will fill one entry of the table shown in Table 6.

3.3.3 Voicing Arrangement

The nature of each of the two overlapping speech signals in a given frame of co-channel speech can vary in voicing. That is, sometimes the frame contains voiced speech

version	SNR
i	0 dB
ii	6 dB
iii	12 dB
iv	18 dB

Table 5: Different SNR versions of the (0 dB VVR) co-channel signals.

Sentence	SNR				avg
	0 dB	6 dB	12 dB	18 dB	
A	–	–	–	–	–
B	–	–	–	–	–
C	–	–	–	–	–
D	–	–	–	–	–
E	–	–	–	–	–
avg	–	–	–	–	

Table 6: Table of D_{avg} scores to be obtained by testing under different SNRs (with VVR fixed at 0 dB).

on voiced speech (V/V), voiced speech on unvoiced speech (V/UV), or unvoiced speech on unvoiced speech (UV/UV). We have chosen to refer to such differences as different *voicing arrangements*. In that report, we stated that it is necessary to ensure that the recordings to be used for testing contain frames in which each of these three voicing arrangements is represented. However, we do not feel that it is necessary to test each of these three cases explicitly. Each performance measurement D_{avg} of the separation algorithm, as measured in the experiments of the previous two sections, is an implicit average of the performance scores on each of the three voicing arrangements (V/V, V/UV, UV/UV), weighted by the relative frequency of those arrangements. As such, we believe that it is reasonable to consider these performance measures, with their implicit weighted-average, as an adequate measure of the performance of the separation algorithm. No explicit testing of each of the three individual voicing arrangements will be conducted at this time.

Silence Detection

The silence detection algorithm used in these experiments is premised on the assumption that the energy level during periods of silence is relatively constant over the duration of a recording, and that the variance of the energy over frames of non-silence (e.g. speech) is greater than the variance of the energy over frames of silence. The procedure begins by breaking the recording to be analyzed into non-overlapping adjacent frames of 32 ms each. For each frame, the energy is computed and stored. Then, a histogram is computed on the energy measures for all the frames. The histogram bin with the most counts is taken to be the one indicating frames of silence. In the case of a tie, the histogram bin with the lower corresponding energy measure is chosen. Finally, all frames with energy measures less than or equal to the upper (energy) boundary of the bin with the maximum number of counts are considered to contain silence.

4 Experimental Results

As discussed above in Section 1.3, the pole-partitioning separation algorithm fails to accomplish its intended task of frame-wise speaker separation due to the inability to accurately estimate the polynomials $N_{co}(z)$ and $D_{co}(z)$ in Step 1 of the algorithm. As a result, we have opted not to perform the tests outlined in the **Test Plan** document (and repeated in Section 3 of this document), as the results of those experiments would be inconclusive. Instead, what we have done is compiled the collection of five co-channel recordings listed in Table 2. For each of these five sentences, four different versions (i–iv) corresponding to different VVRs, as described in Table 3, have been synthesized. These twenty sentences, corresponding to the entries in Table 4, were then submitted to the pole-partitioning separation algorithm. These files have been saved on computer tape with the filenames listed in Table 7.

We have chosen the Steiglitz-McBride algorithm described in Section 2.3.2 to perform Step 1 of the pole-partitioning algorithm, that of estimating the polynomials $N_{co}(z)$ and $D_{co}(z)$. This choice is somewhat arbitrary, in that none of the currently available ARMA parameter estimation techniques, as discussed in Section 2.3.2, generate sufficiently accurate polynomial estimates for our purposes. Nevertheless, this method was deemed to represent the best compromise between performance and computational complexity among the different methods available.

No quantitative performance measures have been collected for these twenty test co-channel recordings. As discussed above, such results would be inconclusive, and furthermore, they could possibly be misleading. Instead, the forty output files (2 output files per co-channel recording $\times$ 20 co-channel recordings) generated by the pole-partitioning separation algorithm have been saved on computer tape for reference and future analysis. The twenty co-channel recordings and the forty corresponding output files are listed in Table 8.

5 Discussion/Conclusions/Future Work Recommendations

We summarize the results of our findings as follows. A new co-channel speaker separation algorithm, for performing separation on a frame-by-frame basis, has been

recording #	file name	sentence 1 index #	sentence 2 index #	VVR
1	cochan.A1.dat	1	2	0 dB
2	cochan.A2.dat	1	2	6 dB
3	cochan.A3.dat	1	2	12 dB
4	cochan.A4.dat	1	2	18 dB
5	cochan.B1.dat	3	4	0 dB
6	cochan.B2.dat	3	4	6 dB
7	cochan.B3.dat	3	4	12 dB
8	cochan.B4.dat	3	4	18 dB
9	cochan.C1.dat	5	6	0 dB
10	cochan.C2.dat	5	6	6 dB
11	cochan.C3.dat	5	6	12 dB
12	cochan.C4.dat	5	6	18 dB
13	cochan.D1.dat	7	8	0 dB
14	cochan.D2.dat	7	8	6 dB
15	cochan.D3.dat	7	8	12 dB
16	cochan.E4.dat	7	8	18 dB
17	cochan.E1.dat	9	10	0 dB
18	cochan.E2.dat	9	10	6 dB
19	cochan.E3.dat	9	10	12 dB
20	cochan.E4.dat	9	10	18 dB

Table 7: Co-channel recordings for experiments and corresponding VVRs and constituent single-speaker recordings (Index numbers correspond to entries in Table 1).

recording #	input file name	output file 1	output file 2
1	cochan.A1.dat	out1.A1.dat	out2.A1.dat
2	cochan.A2.dat	out1.A2.dat	out2.A2.dat
3	cochan.A3.dat	out1.A3.dat	out2.A3.dat
4	cochan.A4.dat	out1.A4.dat	out2.A4.dat
5	cochan.B1.dat	out1.B1.dat	out2.B1.dat
6	cochan.B2.dat	out1.B2.dat	out2.B2.dat
7	cochan.B3.dat	out1.B3.dat	out2.B2.dat
8	cochan.B4.dat	out1.B4.dat	out2.B4.dat
9	cochan.C1.dat	out1.C1.dat	out2.C1.dat
10	cochan.C2.dat	out1.C2.dat	out2.C2.dat
11	cochan.C3.dat	out1.C3.dat	out2.C3.dat
12	cochan.C4.dat	out1.C4.dat	out2.C4.dat
13	cochan.D1.dat	out1.D1.dat	out2.D1.dat
14	cochan.D2.dat	out1.D2.dat	out2.D2.dat
15	cochan.D3.dat	out1.D3.dat	out2.D3.dat
16	cochan.E4.dat	out1.D4.dat	out2.D4.dat
17	cochan.E1.dat	out1.E1.dat	out2.E1.dat
18	cochan.E2.dat	out1.E2.dat	out2.E2.dat
19	cochan.E3.dat	out1.E3.dat	out2.E3.dat
20	cochan.E4.dat	out1.E4.dat	out2.E4.dat

Table 8: Twenty co-channel recordings and forty corresponding output files.

proposed. The algorithm, called the **pole-partitioning separation algorithm** consists of two steps: 1) estimation of the polynomials $N_{co}(z)$ and $D_{co}(z)$ of an ARMA model $S_{co}(z) = \frac{N_{co}(z)}{D_{co}(z)}$, and 2) estimation of two independent AR models $S_1(z) = \frac{\sigma_1}{A_1(z)}$ and $S_2(z) = \frac{\sigma_2}{A_2(z)}$ from these polynomials $N_{co}(z)$ and $D_{co}(z)$. The *blind* estimation of the parameters of an ARMA process, i.e. estimation when only the output signal is available, as required in Step 1 is a problem unto itself. We note that this problem is different than the typical *system identification* problem, in which both the output signal and the input signal are available. Several different methods exist for performing (blind) ARMA parameter estimation, but we have found that while such methods are often adequate for signal *modeling*, they are insufficient for parameter *estimation*. When we rely on any of these methods to estimate the polynomials $N_{co}(z)$ and $D_{co}(z)$, the subsequent Step 2 fails to properly estimate the single-speaker models $S_1(z)$ and $S_2(z)$. However, if $N_{co}(z)$ and $D_{co}(z)$ are known a priori, as in the case of co-channel recordings synthesized from multiple single-speaker recordings, or alternatively, if $N_{co}(z)$ and $D_{co}(z)$ could be *accurately* estimated, then Step 2, which is performed via the **pole-partitioning algorithm**, can successfully estimate the single-speaker models $S_1(z)$ and $S_2(z)$.

In light of these findings, we have been investigating ways to improve our separation algorithm so as to overcome its current limitations. A seemingly obvious first direction might be to attempt to devise a new ARMA parameter estimation technique which would provide the high accuracy necessary for the pole-partitioning algorithm in Step 2 to generate meaningful results. The task of devising such a method, obviously, is a non-trivial undertaking, the demands of which may exceed those of developing the rest of the co-channel speaker separation system. However, a factor which might be utilized, in order to reduce the complexity of this task, is the fact that in our situation, we are not attempting to estimate an *arbitrary* ARMA model, but one *with a high degree of structure*. That is to say, in the general ARMA estimation task, the numerator and denominator polynomials of the model $H(z) = \frac{N(z)}{D(z)}$ are independent. In our case, however, $N(z)$ and $D(z)$ have a particular form, namely $N(z) = N_{co}(z) = \sigma_1 A_2(z) + \sigma_2 A_1(z)$ and $D(z) = D_{co}(z) = A_1(z)A_2(z)$. We see that in this case $N(z)$ and $D(z)$ are not independent, but are in fact related, albeit in a complex way. This being the case, it might be possible in future work to develop a new ARMA estimation technique, or modify an existing one, to incorporate the constraints imposed by this particular structure so as to obtain more accurate estimates.

In the development of the pole-partitioning separation algorithm, we have implic-

itly assumed the presence of two speakers' voices in each analysis frame of the input co-channel signal. Even if we are guaranteed that there would be at most two different speakers speaking on a given communications channel, this does not guarantee that every input frame will contain two voices. Depending on whether or not each speaker is speaking at a given moment in time, a given input frame may contain: 1) two speakers speaking, 2) one speaker speaking, or 3) silence. Typically, one would want to handle each of these three cases separately. In the case of a frame of silence, the co-channel speaker separation system should pass the silence through unaltered, or perhaps mute the output, rather than attempting to separate the silence frame into two frames of speech. In the latter case, the result would be wasted computations at best. However, since inter-frame information is often utilized in the processing of an input stream, the spurious and misleading estimates generated by attempting to separate silence into two speech signals might very well lead to catastrophic failure of a system. Similarly, a co-channel speaker separation system should not try to separate a frame containing a single speaker's voice into two output frames of speech, for much of the same reasons. Finally, an input frame containing two speakers' voices should be passed on to the pole-partitioning separation algorithm. To address this issue of different voicing arrangements in a given frame of co-channel speech, we had begun developing a method for classifying a given input frame into one of the three cases listed above. Further work is necessary along these lines to develop a working algorithm which would precede the frame-wise separation stage in a co-channel speaker separation system.

A final issue that need be addressed in the future is that of reassembling frames of separated speech into speaker-continuous utterances. For every input frame of co-channel speech, the frame-wise separation stage generates two output frames. However, the separation algorithm does not "know" which speaker each output frame belongs to; it simply generates two separate signal estimates. Thus, means for reassembling these individual separated frames back into speaker-continuous utterances must be developed. This would consist of determining which speaker each output frame belonged to. Some approaches might utilize the assumptions of continuity of pitch contours or spectral envelope across frames in order to make this assignment of frames. Future development is needed along these lines to develop a working algorithm which would follow the frame-wise separation stage in a co-channel speaker separation system.

A schematic of how the different processing stages would fit into the final system is shown in Fig. 1.
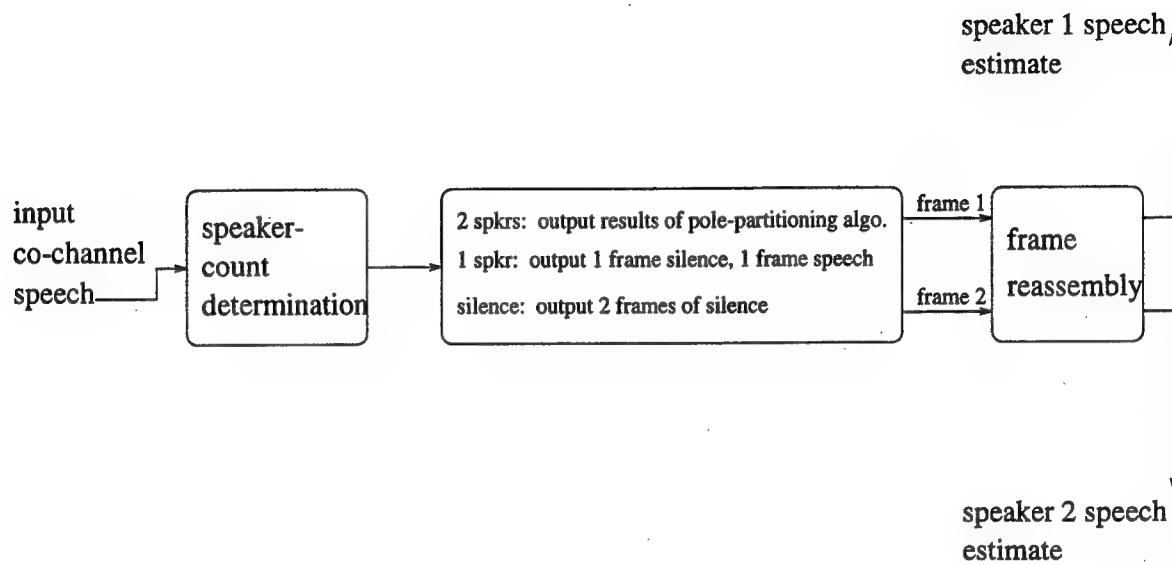
Figure 1: Schematic diagram showing arrangement of (as yet to be developed) processing stages.

References

[1] Jr. Augustine H. Gray and John D. Markel. Distance measures for speech processing. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP-24(5):380–390, 1976.

[2] F. Itakura. Minimum prediction residual principle applied to speech recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP-23:67–72, 1975.

[3] S. Lawrence Marple Jr. *Digital Spectral Analysis with Applications*. Prentice-Hall, 1987.

[4] Gary E. Kopec and Marcia A. Bush. An lpc-based spectral similarity measure for speech recognition in the presence of co-channel speech interference. In *Proceedings ICASSP-1989*, pages 270–273, 1989.

[5] Gary E. Kopec, Alan V. Oppenheim, and Jose M. Tribolet. Speech analysis by homomorphic prediction. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP-25(1):40–49, 1977.

[6] National Institute of Standards and Technology. Darpa timit acoustic-phonetic continuous speech corpus. CD-ROM. [software].

[7] Lawrence R. Rabiner and Ronald W. Schafer. *Digital Processing of Speech Signals*. Prentice-Hall, 1978.

[8] K. Steiglitz. On the simultaneous estimation of poles and zeros in speech analysis. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP-25:229–234, 1977.

[9] K. Steiglitz and L. E. McBride. A technique for the identification of linear systems. *IEEE Transactions on Automatic Control*, AC-10:461–464, 1965.

[10] Charles W. Therrien. *Discrete Random Signals and Statistical Signal Processing*. Prentice-Hall, 1992.

[11] Charles W. Therrien and Carlos H. Velasco. An iterative prony method for arma signal modeling. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 43(1):358–361, 1995.

A   Review of Previous Work

The appendix attached on the following pages is the Master's thesis of one of the graduate student research assistants employed under this contract. Any questions regarding this thesis may be addressed to: Alvin Garcia: phone: (908)445-0573, e-mail: `alvin@caip.rutgers.edu`.

APPENDIX A

METHODS FOR CO-CHANNEL SPEECH SEPARATION

BY ALVIN A. GARCIA

A thesis submitted to the

Graduate School—New Brunswick

Rutgers, The State University of New Jersey

in partial fulfillment of the requirements

for the degree of

Master of Science

Graduate Program in Electrical Engineering

Written under the direction of

Professor Richard J. Mammone

New Brunswick, New Jersey

May, 1995

Methods for Co-channel Speech Separation

by Alvin A. Garcia
Thesis Director: Professor Richard J. Mammone

It is a widely appreciated fact that when a given speech signal has been corrupted by the superposition of the voice of an interfering speaker, the result is a co-channel speech signal with markedly decreased intelligibility of both the desired talker's speech, as well as that of the interfering speaker. Furthermore, such contaminated signals result in severe performance degradation of automatic speech processing systems, such as speech recognizers, speaker identification/verification systems, and speech coders. The goal of a Co-channel Speaker Separation (CCSS) system is to separate the voices of a number $n > 1$ of individual speakers from a single composite signal consisting of the sum of the voices of the n speakers.

In this thesis, a comprehensive review is conducted of all the major systems proposed to date for performing co-channel speaker separation. It will be shown that most systems are designed to perform separation of only voiced speech and, as such, processing can be divided into two logical steps: 1) estimation of the fundamental frequency or pitch of each speaker's voice, and 2) separation of the two voice signals.

A new method is proposed for such multi-pitch estimation from the co-channel signal. The performance of the method is experimentally evaluated and compared with the pitch estimation algorithms of the previous work under a variety of testing conditions. It is shown that the new method offers the best performance in the case of

clean speech, and second best performance in the case of channel degradations. In the case of additive noise, however, performance dropped substantially.

Acknowledgements

I am grateful to my advisor, Dr. Richard J. Mammone, for his guidance and patience in my preparation of this thesis, and for helping make this research effort possible. I would also like to thank the other members of my thesis committee, Dr. Joseph Wilder and Dr. Peter Meer, for the invaluable knowledge I have learned from them in the classroom and in the laboratory. Finally, I would also like to thank my research colleagues and seniors at CAIP Center for their valuable suggestions and insights and great sense of humor.

I am indebted to my parents for their love and encouragement, and for the sacrifices they have made for me. I am also grateful to my sister, Eileen, for her love and support.

Dedication

This effort is dedicated to my parents, Esteban and Josefina Garcia.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

ACF   is for Auto Correlation Function

AGC   is for Automatic Gain Control

AR   is for Auto Regressive

AMDF   is for Average Magnitude Difference Function

ARMA   is for Auto Regressive Moving Average

CCSS   is for Co-Channel Speaker Separation

dB   is for deciBel

FFT   is for Fast Fourier Transform

FIR   is for Finite Impulse Response

FM   is for Frequency Modulation

FTT   is for Fast Triangular Transform

LPC   is for Linear Predictive Coding

ML   is for Maximum Likelihood

SNR   is for Signal to Noise Ratio

STFT   is for Short Time Fourier Transform

TIR   is for Target to Interference Ratio

VVR   is for Voice to Voice Ratio

Chapter 1

Introduction

1.1 Problem Description

Often times in communications scenarios, the voice transmission of a given speaker is corrupted by the voices of other, interfering, speakers. The interference may be introduced in the communications channel, at some point between the transmitting and receiving ends. Such a situation would arise where there are multiple transmitters on a common communications channel, as is sometimes the case in tactical communications systems. Alternatively, such interference may be introduced at the transmission site itself, rather than along the communications path. This would be the case if, for instance, the microphone at the transmitting end was not acoustically isolated, in which case all background noises, including voices, would be transmitted along with the intended (voice) transmission. This situation is exemplified by speaker-telephones and other hands-free communications devices. Regardless of the actual cause of this interference, the result at the receiving end is a co-channel speech signal — a composite signal consisting of the sum of the voices of multiple people speaking at the same time.

In such co-channel signals, the voice of both the intended (target) speaker and the interfering speaker(s) are marked by decreased intelligibility. The human brain can compensate for such interference when *binaural* data is available, as is often the case when listening to one person speaking in a room where many people are talking. However, when such interference occurs over a monaural channel, separation of the different voices by human listeners is much more difficult [30]. Additionally, such co-channel speech adversely affects the performance of many automatic speech processing systems, such as speech recognition systems [14] and speaker identification systems.

The goal of an automatic **co-channel speaker separation** (CCSS) system is to

take a co-channel speech signal and separate it into the constituent speakers' voices. The resulting separated signals would then be available for human consumption or for input into some type of automatic speech processing system.

1.2 Practical Considerations and Limitations

An *ideal* CCSS system would be able to perfectly reconstruct each of the N constituent voice signals from the co-channel signal. In other words, if we denote the voice signal of the ith speaker by $s_i[n]$ and the co-channel signal by $s_{co}[n]$, then

$$s_{co}[n] = \sum_{i=1}^{N} s_i[n]$$

and from this signal, the ideal system would be able to exactly retrieve all N constituent signals $\{s_1[n], \ldots, s_N[n]\}$. Note that the construction of such a system is impossible, even for $N = 2$. If it *was* possible, then it would be possible to transmit an arbitrary number $N > 1$ of independent signals, each having bandwidth W, along a communications channel of bandwidth W, thereby violating some of the basic laws of communications and information theory. Therefore, *no* CCSS system will be able to exactly reconstruct the individual voice signals from the co-channel signal, even for the relatively "simple" case of $N = 2$. In light of this limitation, however, there are two considerations which allow for at least the potential of some degree of success in CCSS systems. First of all, *exact* reconstruction of the speech signals is not necessary for effective separation. What is necessary is that the reconstructed signals are *perceptually* similar to the original signals. Exactly *what* constitutes "perceptual similarity" will depend on whether the final consumer of the reconstructed speech is human or machine. Furthermore, if the reconstructed speech is to be input to a machine, then the measure of perceptual similarity will depend on the particular parameterization (LPC, cepstral coefficients, or otherwise) being used to represent the speech signals. The second consideration is that speech signals are constrained by the nature of the human speech production system. Knowledge of these constraints, and their implications for the produced speech signals, allows for the reduction of the CCSS problem from the general task of recovering *arbitrary* signals from a co-channel signal to the restricted
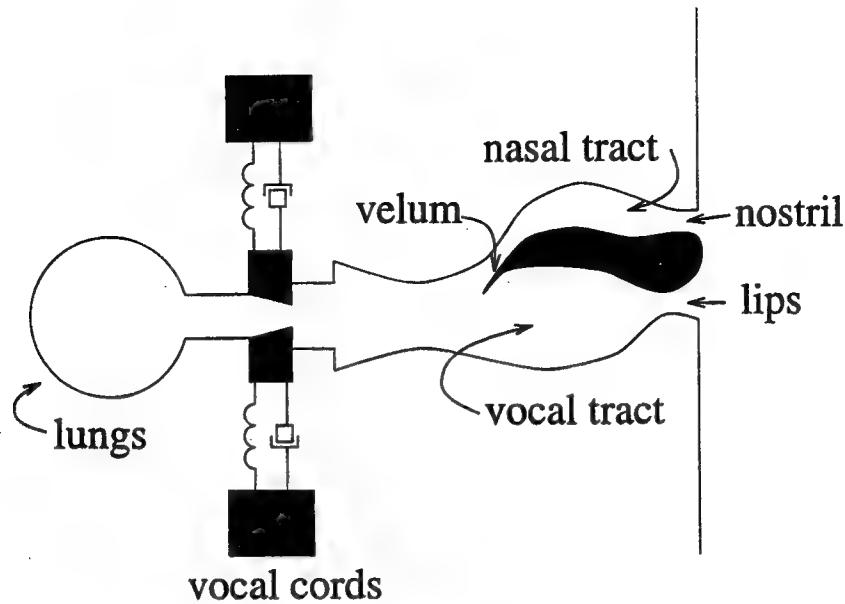
Figure 1.1: Schematic diagram of the human vocal system (after Flanagan et al.)

task of recovering signals, with some known characteristics, from the co-channel signal.

1.3 Background

1.3.1 The Mechanism of Speech Production and the Nature of Speech Signals

The human vocal system is comprised of the vocal tract, the nasal tract, and the lungs [32]. The vocal tract consists of the pharynx and the oral cavity, extending from the glottis (vocal cords) to the lips. The nasal tract extends from the velum to the nostrils. The vocal system is represented schematically in Fig. 1.1 [8]. The sounds of speech are the acoustic waves produced by this system when air from the lungs is forced through the vocal and nasal tracts. As speech is produced, the nasal tract may intermittently be closed off from the vocal tract by means of the velum, and the vocal tract itself will change in shape all along its length. These variations allow for the production of different sounds.

Roughly speaking, most speech sounds can be classified as *voiced* or *unvoiced* [32].[1]

---

[1]Strictly speaking, speech sounds can also belong to a third class, *plosive* sounds. However, most
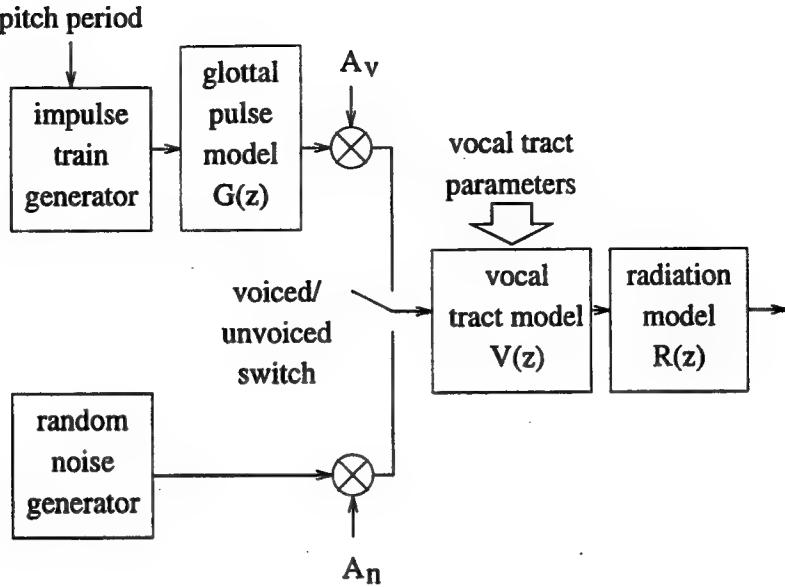
Figure 1.2: General model for speech production (after Rabiner 1978.)

Voiced sounds are created when the vocal cords are made to oscillate, causing a quasi-periodic air wave to excite the vocal tract. The resulting speech waveform is therefore, also quasi-periodic. *Unvoiced* sounds, or *fricatives*, are produced when air is forced pass a constriction made in the vocal tract, causing turbulent air flow which introduces a random noise excitation signal into the vocal tract. In this case, the resulting speech waveform exhibits no periodicity. In either case, the vocal/nasal tract system exhibits resonances, known as *formants*, at different frequencies.[2] The frequency locations and bandwidths of the formants change over time as the shape of the vocal tract and the position of the velum change during the production of speech. This frequency selectivity results in the shaping, or filtering, of the spectrum of the excitation signal.

1.3.2   Speech Models

A commonly-employed general discrete-time model for speech production is shown in Fig. 1.2 [32]. Here $V(z)$ represents the Z-transform of the discrete-time filter which

---

speech sounds fall into one of the other two categories.

[2]Sometimes, *anti*-resonances are present in the frequency response of the vocal/nasal tract system, such as during the production of *nasals*.

models the frequency response of the vocal/nasal tract system. $R(z)$ models the effects of radiation of the acoustic wave outwards from the lips, and $G(z)$ is the impulse response representation of one cycle of the glottal pulses which are produced during voiced speech. The parameters A_v and A_n represent the gain factors for the voiced and unvoiced excitation sources, respectively, and the switch toggles between the two sources, depending on whether or not the modeled speech is voiced or unvoiced. During the production of speech, the physical components of the vocal system, corresponding to the parameters $G(z)$, $V(z)$, $R(z)$, A_v, A_n, "pitch period", and "voiced/unvoiced" switch, move relatively slowly over time. As such, it is often assumed, in both analysis and synthesis of speech signals, that these model parameters are constant over short time intervals on the order of 10–20 milliseconds. Furthermore, often times the transfer functions of the vocal tract model, radiation model, and glottal pulse model in the case of voiced speech, are lumped together into a single transfer function model:

$$H(z) = G(z)V(z)R(z) \tag{1.1}$$

In general, this resulting composite transfer function $H(z)$ may be an autoregressive moving average (ARMA) filter. Typically, however, $H(z)$ is modeled as a strictly autoregressive (AR) filter, whose parameters are determined by some linear predictive coding (LPC) analysis of the speech signal.

As mentioned above, the excitation signal for voiced speech is quasi-periodic. Therefore the corresponding magnitude spectrum is roughly a line spectrum with peaks situated at integral multiples of the fundamental frequency f_0, or pitch.[3] For the random noise excitation signal of unvoiced signals, the corresponding magnitude spectrum is roughly "white" or flat, and exhibits no evident fine structure, as with voiced signals. When the excitation signal propagates through the vocal/nasal tract system, the resulting signal has a spectrum whose gross spectral shape is dictated by the frequency selectivity of that system, and whose fine spectral detail is governed by the type of excitation signal. This is illustrated in Figs. 1.3 and 1.4. Fig. 1.3 shows a sample spectrum

---

[3]In the ensuing discussion, the terms "pitch" and "fundamental frequency" will be used interchangeably, although strictly speaking, usage of the term "pitch" refers to the *perception* of the fundamental frequency f_0.

of a (synthetic) periodic excitation signal, the magnitude frequency response $|H(\omega)|$ of the vocal/nasal tract, and the spectrum of the voiced speech signal resulting when the vocal tract filter is driven by the periodic excitation signal. Fig. 1.4 shows a sample spectrum of a random, non-periodic excitation signal, the same magnitude frequency response $|H(\omega)|$ of the vocal/nasal tract, and the spectrum of the resulting unvoiced speech signal.

1.4 Outline of the Thesis

In Chapter 2, a comprehensive review of all major previous work on the co-channel speaker separation task is conducted. Limitations of each of the proposed methods are discussed. It is shown that the operation of most of the methods can be divided into three logical steps: 1) fundamental frequency estimation from the co-channel signal, 2) separation of the signals using these f_0 estimates, and 3) association of the separated frames with the different speakers, so as to assemble these frames into continuous speech utterances.

A new method for fundamental frequency estimation from the co-channel signal is also presented in Chapter 2. Its operation is discussed and various implementational considerations are addressed.

In Chapter 3 the performance of the new method is evaluated and compared with the pitch estimation techniques of the CCSS methods reviewed in Chapter 2. A discussion and analysis of these results follows in Chapter 4.
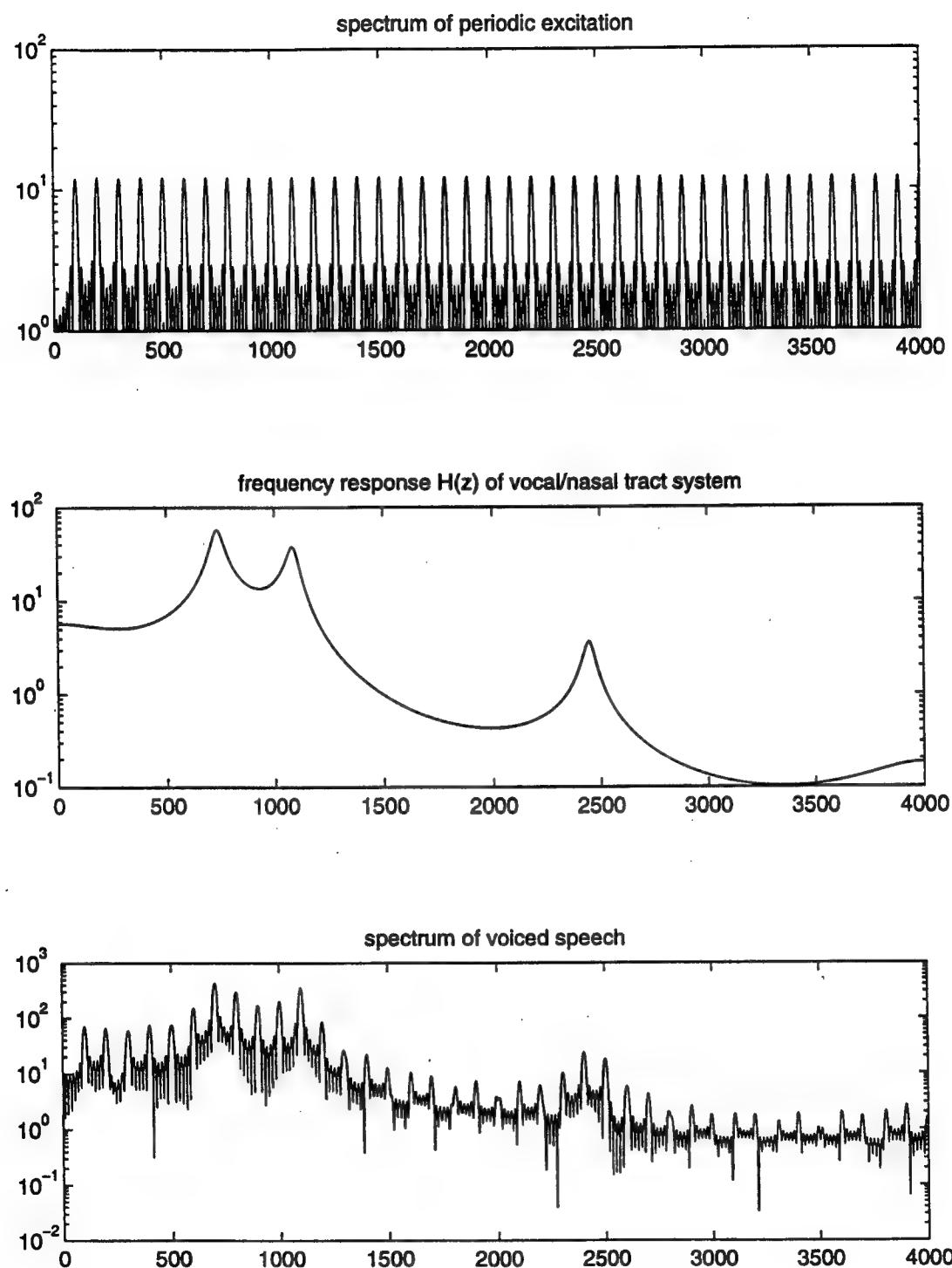
Figure 1.3: Illustration of the spectra of voiced speech.
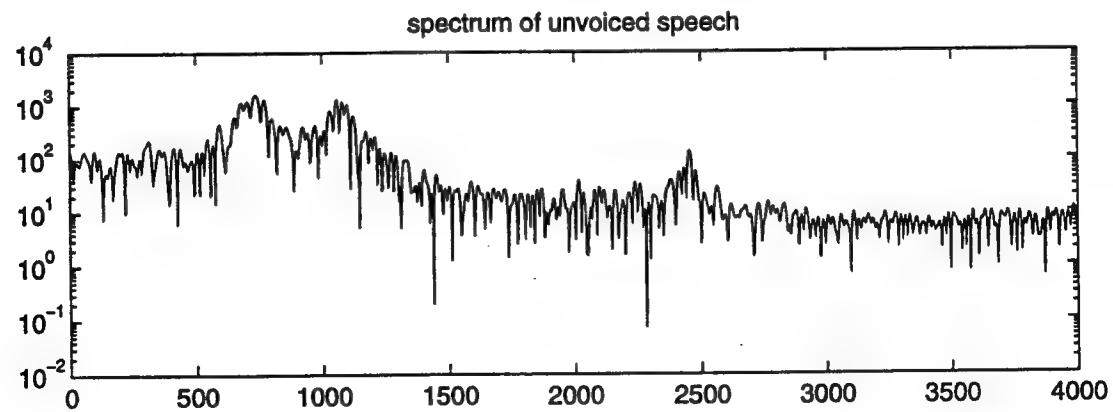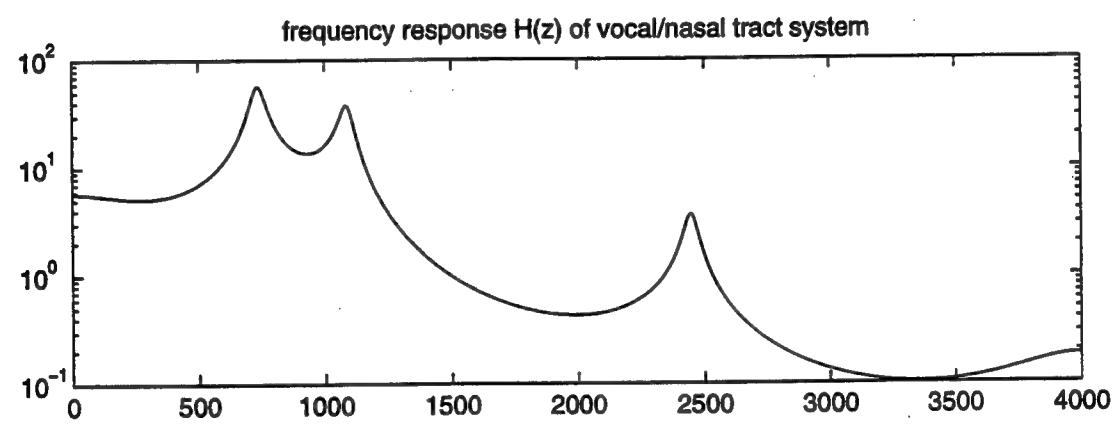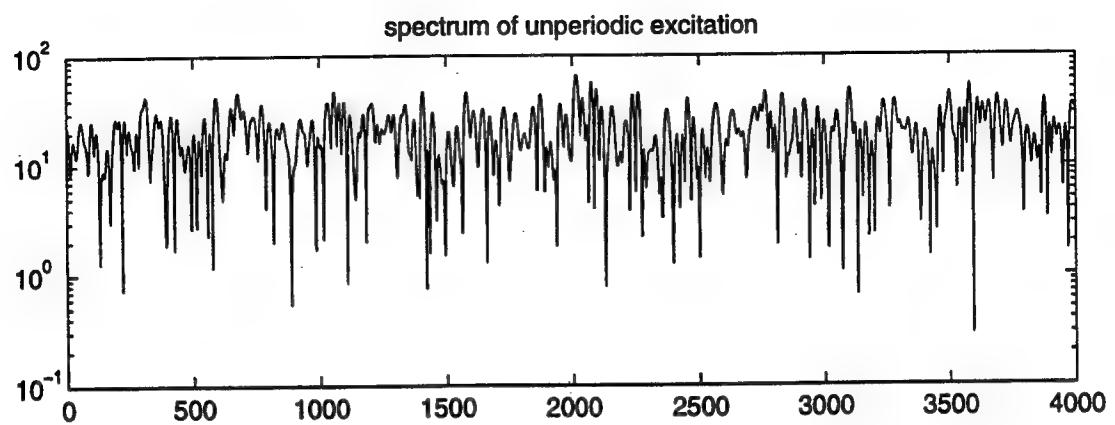
Figure 1.4: Illustration of the spectra of unvoiced speech.

Chapter 2

Review of Co-channel Speaker Separation Systems

2.1 Speaker Separation Algorithms

Over the years, a number of methods have been proposed for handling the problem of co-channel speech speaker separation; reference [6] contains a brief review of many of these methods. While many of the CCSS algorithms can theoretically handle multiple speakers, in practice only the two speaker case has been addressed.[1] Typically, one speaker is considered the desired or "target" speaker, while the other speaker is referred to as the "interfering" speaker; the ratio of the target speaker's power to the interfering speaker's power is sometimes referred to as the Target to Interference Ratio (TIR) or Voice to Voice Ratio (VVR). In most CCSS systems to date, the processing can be divided into three main components: 1) fundamental frequency estimation, 2) actual separation of the target and interfering speakers' speech, and 3) association of the separated speech segments with the appropriate speaker. Many approaches only address steps 1) and 2). The last processing stage is necessitated by the fact that in many algorithms, the processing of steps 1) and 2) is achieved frame-wise. That is, the input co-channel signal is divided into short frames of contiguous samples and, for each of these input frames, two output frames are generated, one corresponding to the target speaker's voice and the other corresponding to the interfering speaker. However, the processing involved in the separation typically doesn't "know" which is the target speaker and which is the interfering speaker; it only assumes the presence of two voices. Therefore, once this separation is accomplished, some type of post-processing is necessary to maintain continuity of speaker identity from (output) frame to frame. This

---

[1]In [25], Min et al. suggest possible extensions of their method to the case of three speakers; however, at the time of the paper's publication, no experiments had been conducted to this end.

processing is still necessary, even if the separation processing doesn't proceed frame-wise, if there are periods of silence in the co-channel signal. In this case, continuity in speaker identity must be maintained across the silence intervals. Finally, it should be noted that some of the methods do not attempt reconstruction of the separated signals. Rather, they segregate the components of the two speech signals in the representation space of some signal model, such as a correlogram for instance, but do not provide a means for reconstructing the corresponding time domain signals from this representation space.

Most methods are designed to handle only *voiced* speech. That is, both the target and interfering voices are assumed to be voiced. A crucial processing step in these algorithms is the accurate determination of the target speaker's pitch, the interfering speaker's pitch, or both pitches simultaneously. Once the pitch of each of the two speakers is estimated, each of the component speech signals can be extracted by comb-filtering the co-channel signal, or by analogous processing techniques, such as synthesis from selective reconstruction of the co-channel spectra. Alternatively, some methods only estimate the pitch of *one* speaker and then extract his/her voice via comb-filtering or similar spectral enhancement techniques. The voice of the other speaker is then recovered by "subtracting"[2] the estimate of the first speaker's voice from the composite signal, leaving behind the second speaker's voice.

As outlined above, most proposed CCSS algorithms perform frame-wise separation of the co-channel speech; a few also address the task of associating the separated speech segments. However, a number of equally important issues have yet to be adequately addressed before any of these algorithms can be utilized in a practical CCSS system. First is the ability to determine the actual number of people speaking at each instant of time. Even in the two-speaker case, both speakers aren't always speaking; sometimes one person, or both people, are silent. Few of the CCSS methods to date address this critical issue; this is a severe limitation, as pauses of silence occur naturally in normal speech. A practical CCSS system needs to employ *input-dependent* speech enhancement,

---

[2]The quotes indicate that subtraction isn't necessarily performed on the signals themselves, but may be performed on some *representation* of the signals, such as a Fourier Transform, for instance.

in the sense that the type of processing, and the decision as to whether or not *any* processing should be done at all, should depend on the nature of the input signal (see [2], for instance). For example, in the case of only one speaker, the system should not attempt to make some reconstruction of the silent speaker's voice. At best, the result would be wasted computations; more likely, the resulting processing of silence would lead to a meaningless reconstruction, which may not only result in a confusing output signal for that analysis interval, but may also interfere with subsequent processing stages which rely on the estimates from previous frames.

A second task which remains to be developed further is that of detecting whether the input signal(s) are voiced or unvoiced. Again, the processing of the co-channel signal should proceed differently, depending on the voicing[3] of the input speech. In the case of a single, unvoiced speech signal present in the co-channel signal, a CCSS system might pass the signal through unprocessed, whereas in the case of a single, voiced speech signal, some type of enhancement by comb-filtering might be employed. In the case of voiced/unvoiced speech and unvoiced/voiced speech, the CCSS system should process the unvoiced speech differently than the voiced speech; e.g. comb-filter the co-channel signal to enhance the voiced speech and use a multi-notch filter on the co-channel signal to suppress the voiced speech and leave the unvoiced speech.

The case of unvoiced/unvoiced speech is a third issue which needs to be addressed. The author is not aware of any methods to date which have been designed to handle separation of unvoiced/unvoiced speech. In this case, there are no regularities in the time waveforms or spectra of either speaker's speech which can be used to segregate the two speech signals. This area remains open for exploration.

2.2   Previous CCSS Work

2.2.1   Parsons' Method of Harmonic Selection (1976)

One of the earliest efforts towards the co-channel speaker separation task was by Parsons in 1976 [30]. The basis of the separation process involves identification and association

---

[3]By *voicing*, we mean the nature of the speech signal excitation: *voiced* or *unvoiced*.

of harmonic peaks in the magnitude spectrum of the co-channel signal, which is assumed to consist of only voiced speech. The operation of the entire system can be broken down into the following stages: 1) pre-processing, 2) peak separation, 3) pitch extraction, 4) tracking, and 5) re-construction. Pre-processing consists of blocking the input signal into Hanning-windowed frames, and then computing the corresponding STFT (short time Fourier transform), hereafter "spectrum," of each frame. In the peak separation stage, local maxima in the magnitude spectrum are identified as potential harmonic peaks, and estimates of their frequency, amplitude, and phase are entered into a "peak table." Overlapping, or superimposed, peaks are detected by testing each peak in the magnitude spectrum for symmetry, sufficient distance from neighboring peaks, and well-behaved phase. If overlapping peaks are detected, they are separated by subtracting an ideal peak shape, computed from the window shape and the estimated parameters (FM rate, amplitude) of the larger peak, from the spectrum, leaving behind the smaller of the two overlapping peaks, whose frequency and amplitude estimates are then entered into the peak table. Pitch extraction is accomplished by employing an adaptation of the Schroeder histogram method [34] on the frequencies gathered in the peak table. In the Schroeder histogram method, a histogram is formed of the frequencies of all peaks in a given signal's magnitude spectrum, and all integer sub-multiples of these frequencies. Since harmonic frequencies are integral multiples of the fundamental frequency, the maximum in the Schroeder histogram should correspond to the fundamental frequency of the signal. Parsons adapted this method for dual pitch estimation as follows: First the Schroeder histogram is applied to the frequencies present in the peak table to identify the pitch of the louder speaker. Then this pitch and all integral multiples of this pitch are removed from the peak table. The Schroeder histogram is then applied to the peak frequencies remaining in the peak table, which presumably belong to the quieter speaker, and so the maximum of the resulting histogram should correspond to the pitch of the quieter speaker. Once pitch estimates of both speakers are calculated in this manner, the individual peaks in the peak table are associated with the speaker with a harmonic frequency, i.e. integral multiple of f_0, closest to the frequency of that peak. An output frame for each speaker is generated by computing the inverse STFT of the

spectrum consisting of only those peak components attributed to that speaker. At this point, the actual task of separation is finished, and what remains is the association of the separated segments of speech with the corresponding speakers. Parsons prescribes a method to accomplish this association task which utilizes the assumption of continuity of the pitch track (trajectory) for each speaker. At each step, the pitch of each of the two output frames is compared with the pitch predicted by a least squares linear fit of the previous three pitch estimates of each speaker's pitch track. Each output frame is then assigned to the pitch track whose predicted pitch for that time instant is closest to that frame's actual pitch. This process is repeated until pitch tracks are generated for the entire duration of the co-channel utterance.

Reported performance of this system is "generally good," performing best when both speakers are of comparable loudness. Nevertheless, there are a number of practical difficulties which can potentially hinder performance of this system. First, the system relies on the assumption that both speakers' speech is voiced; in the event of frames where one or both speakers' speech is unvoiced, the resulting pitch estimates are meaningless, and the resulting time-domain waveform reconstructions are not representative of the actual individual speech signals. Furthermore, such spurious pitch estimates would degrade the subsequent pitch tracking. Another limitation is that the system assumes that the underlying speech is continuous; i.e. that there are no pauses of silence. In the event of silence intervals in either speaker's speech, the method proceeds to make spurious pitch estimates, as in the case of unvoiced speech. Reliance on these assumptions is not unique to Parsons' algorithm; as stated earlier, these problems are common to most CCSS algorithms. However, there remain a number of limitations specific to Parsons' method. First is the issue of sufficient frequency resolution. A time window duration of of 51.2 msec is used, representing a compromise between frequency resolution and time resolution. As is well known, the frequency resolution (Δf) of a Fourier Transform-based spectrum is proportional to the reciprocal of the window duration: $\Delta f \propto \frac{1}{T}$. Infinite frequency resolution ($\Delta f = 0 \Leftrightarrow T = \infty$) would be desired to facilitate the task of resolving and identifying harmonic peaks. However, since the parameters of speech signals remain approximately stationary for only short periods of

time, roughly 20–30 msec, the time window must be limited to these durations, thus limiting frequency resolution. This frequency resolution limitation is exacerbated by the fact that the pitch of speech is never perfectly stationary (constant), even within these short time periods, and in fact can change by up to 10% between two successive periods [22]. For the range of human pitch frequencies, there are typically several pitch periods within an analysis frame of 51.2 msec. Pitch non-stationarity within the analysis frame results in increased FM broadening of the harmonic peaks with increasing frequency, in a constant-Q fashion. Fig. 2.1 shows the magnitude spectrum of a frame of voiced speech which illustrates the broadening of harmonic peaks with increasing frequency. Parsons claims that 75% of the overlapping peaks can be resolved by the
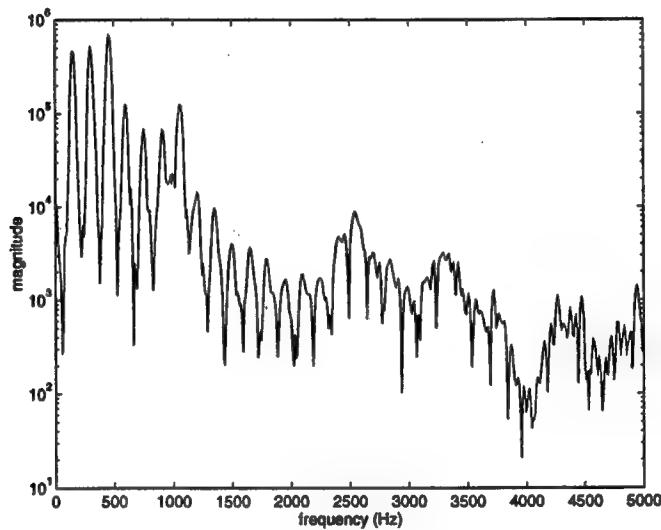
Figure 2.1: Illustration of harmonic smearing due to pitch non-stationarity within analysis interval.

spectral subtraction method described above; the remaining 25% are spaced too closely to be resolved. However, Parsons notes that these peaks, which cannot be uniquely assigned to one or the other speakers and are thus assigned to both speakers, result in crosstalk in the reconstructed speech signals.

Another difficulty encountered in this system occurs when one speaker's voice is

significantly louder than the other's. In these situations, Parsons notes that the intelligibility of the target speaker's reconstructed speech decreases with increased volume of the interfering speaker. This is because as the volume of the interfering speaker increases, the spectral peaks due to the target speaker become more and more obscured, until they can only be detected via the spectral subtraction method detailed above. However, the spectral subtraction method relies on the assumption of ideal harmonic peak shape, and two factors limit its effectiveness, especially in the case of a louder interfering speaker, even if the two speakers' pitches are estimated accurately. First is the presence of noise: if the noise floor is comparable in power to the quieter speech signal, obviously the harmonic peaks of the quieter speaker will be difficult to identify among the peaks in the difference spectrum left after spectral subtraction of the larger peaks. Secondly, the spectral subtraction method computes ideal peak shapes based upon the estimated frequency, amplitude, and FM rate of the larger peaks, and the window shape. These estimates will always be biased by the presence of the smaller, obscured peaks. Furthermore, a *linear* pitch (corresponding to a constant FM rate) is assumed in computing the ideal peak shapes; it is not known how to compute peak shapes due to arbitrary pitch changes, or equivalently, arbitrary frequency modulations [41]. These inaccuracies in peak shape computation also lead to poor estimates of the smaller peaks, leading to poor pitch estimates and reconstruction of the quieter speech signal.

2.2.2 Hanson and Wong's Harmonic Magnitude Suppression Technique (1984)

In [10], Hanson and Wong propose a method for *suppression* of an interfering speaker's voice in a co-channel speaker scenario, rather than for actual *separation*, which would imply recovery of both the target and interfering speaker's speech. The basis of the method is spectral subtraction of the interfering speaker's voice from the co-channel signal's short-time spectra. First, it is assumed that the pitch of the interference can be determined for each frame of the input co-channel signal; the authors do not propose a method for accomplishing such pitch estimation, but maintain that in the case of

negative TIRs, the energy of the co-channel signal is dominated by the interference, and so any pitch estimator should tend to detect the pitch of the interference signal. Once the pitch f_0 of the interference signal is determined, its magnitude spectrum $|\hat{N}(f)|$ is approximated by the spectrum of the linear combination of windowed sinusoids with frequencies $p \times f_0$, $p = 1, 2, \ldots L$, and amplitudes given by the amplitude of the co-channel spectra at those harmonic frequencies. Mathematically, this is given by:

$$|\hat{N}(f)| = \sum_{p=1}^{L} D_p W_{ml}(f - f_p)$$

where D_p is the amplitude of the co-channel spectra at frequency $p \times f_0$ and W_{ml} is the main lobe of the magnitude Fourier transform of the window shape. The estimate of f_0 is refined by perturbing its value within a small interval about the initial estimate, so as to find the value which maximizes the power of $\hat{N}$. The interference magnitude spectrum estimate is then subtracted from the co-channel magnitude spectrum, leaving an estimate of the target signal magnitude spectrum. This difference magnitude spectrum is then converted into a time domain signal via an inverse STFT using the phase estimates from the co-channel spectrum. The algorithm is described diagrammatically in Fig. 2.2.
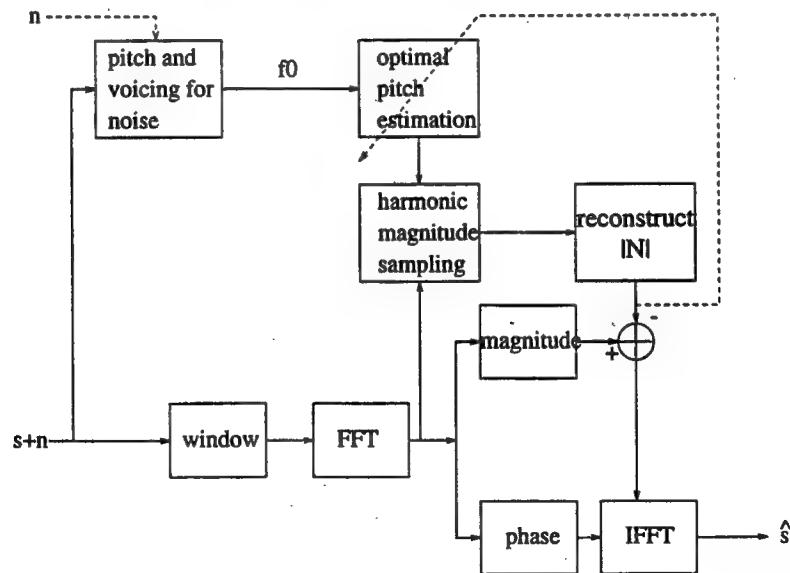
Figure 2.2: Schematic diagram of HMS speech enhancement system.

The system was evaluated in terms of intelligibility of the enhanced target speech vs. intelligibility of the unprocessed co-channel speech; reported performance indicated that the system offered significant improvements. The system, however, suffers a number of practical limitations. First and foremost is the lack of provisions for estimation of pitch from the co-channel signal. This feature is such a fundamental element of most CCSS systems that its omission effectively renders the system unusable for any realistic situation, where the interference and target signals are not known prior to summation. Also, as with Parsons' design, this system does not attempt to detect voicing or silence, nor does it provide accommodations for appropriate handling of unvoiced speech. Furthermore, although the system is designed as a *suppression* rather than a *separation* system, there still remains the problem of maintaining speaker continuity between frames; the system assumes that the pitch estimation algorithm will always find the pitch of the interference speaker and not that of the target speaker. Such an assumption isn't valid, since in general, the relative power of the two speakers' signals will vary. Furthermore, even if the relative power of the speakers does not vary, i.e. the TIR doesn't change sign, there is no guarantee that the pitch estimation algorithm will always pick the pitch of the louder speaker.

Another limitation of the system is the assumption of pitch stationarity within each analysis frame. This assumption limits the amount of interference suppression achievable, even with accurate pitch estimation, because, as indicated in section 2.2.1, pitch non-stationarity results in FM broadening of the harmonic peaks. Since the proposed method assumes stationary pitch, all the peak shapes in the interference spectrum approximation have uniform width, not constant-Q, as is the case for harmonic signals with slightly varying pitch. This is illustrated in Fig. 2.3 which shows the spectra of a 100 msec frame[4] of voiced speech and its strictly periodic approximation, and the resulting difference spectrum. It is evident from this figure that suppression of the signal spectrum is far from complete, due to the mismatch between ideal and actual harmonic peak shapes.

---

[4]Hanson and Wong do not report the duration of the window used; 100 msec is used here for illustration purposes only.
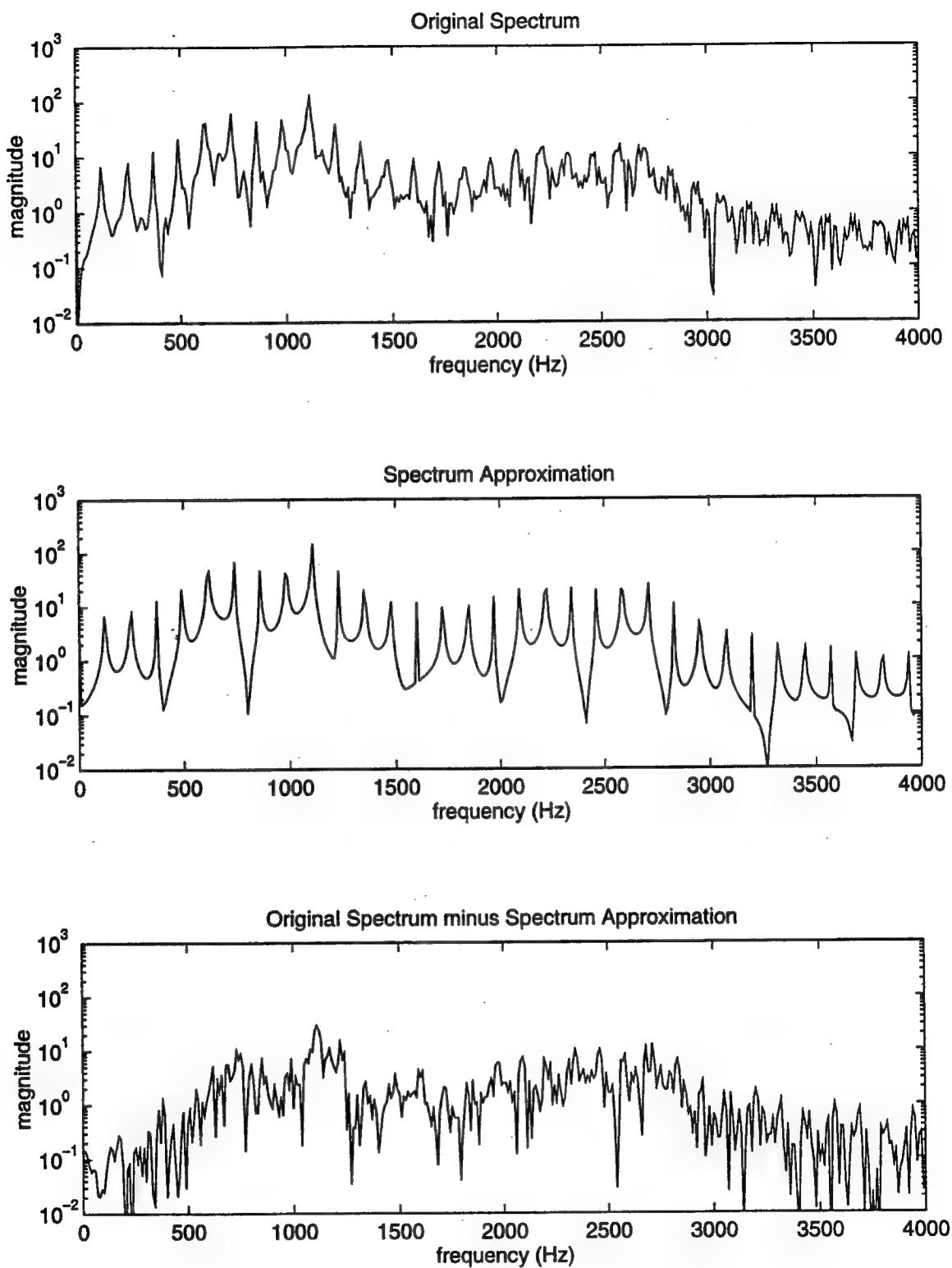
Figure 2.3: Illustration of effects of harmonic smearing in HMS system.

The amount of interference suppression attainable, even with stationary pitch and accurate pitch estimates, is further limited by the use of *magnitude* spectral subtraction, as opposed to complex domain spectral subtraction. The magnitude of the Fourier transform of the sum of two signals will only be equal to the sum of the magnitudes of the Fourier transforms of the individual signals if the two signals are in phase, otherwise they will differ. Since this situation of coherent phase will not occur in general, the estimate of the target signal's spectrum obtained by subtraction of magnitude spectra will not be truly representative of the magnitude spectrum of the actual target signal.

2.2.3   Weintraub's GRASP Sound Separation System (1984)

In [42], Weintraub proposes the GRASP (Grouping Research on Auditory Sound Processing) sound separation system based upon a computational model of the cochlea. The basic operation of the system consists of decomposing the co-channel signal into a cochlear model representation, similar to a filter bank, and then grouping together filter channels which are local in time and which share a "consistent set of features." Weintraub suggests a number of features with which to group filter channels, including: pitch period, pitch dynamics, degree of periodicity, and changes in amplitude. However, the study detailed in the paper only considers a single feature, the pitch period, for lack of a method for measuring consistency along a multiple-feature parameterization of each channel.

In the cochlear model, the incoming signal is passed into a cascade/parallel filter bank arrangement of bandpass filters, with center frequencies spanning the 0–8kHz frequency range. The output of each filter is half-wave rectified and then passed through an automatic gain control, and the pitch of the resulting signal is determined by finding the maximum in its "coincidence function," a function similar to the short-time autocorrelation function. After the pitch is computed for each channel in this way, a histogram is computed of these pitch period estimates, and if a sufficiently large peak is found, a voiced sound is assumed to be present with pitch equal to the frequency corresponding to that peak. All channels with pitch close to this frequency are assigned to a "group." The histogram is then computed on the pitch period estimates of the

remaining unassigned channels, to determine whether or not a second voiced sound is present, and if one is detected, those channels with pitch estimates near the frequency corresponding to this new peak are assigned to a new group. Once a group is started, adjacent time frames are searched for channels with consistent features, i.e. pitch.

The proposed GRASP model was intended only as a portion of an overall framework for speech separation. As such, it lacks many capabilities necessary in a practical speech separation system. First of all, the system does not provide a means for reconstruction of time domain signals, or even magnitude spectra, from the cochlear model representation. Therefore, even if complete source separation was attainable in the cochlear model representation, the corresponding separated sounds could not be heard or even fed into an automatic speech processing system. Obviously, the lack of this important processing stage severely limits the practical utility of the system. More recent work [39] has addressed the reconstruction of time domain waveforms from cochlear model representations. However, such work only allows for reconstruction of time domain waveforms from *complete* correlograms[5]; the GRASP system generates incomplete correlograms because only some channels of the correlogram are assigned to a given sound group, the remaining channels being distributed among the other group(s).

As with most other methods, the GRASP system does not provide a means for separation of unvoiced/unvoiced speech. However, the system does provide means for detection of voicing and estimation of both speakers' pitch from the co-channel signal. In addition, it prescribes a method for maintaining speaker continuity across time based on assumptions of pitch continuity. However, the system requires that there are frequency ranges in which the energy of one of the speakers' voices dominates the output of the filter channels in those ranges. If the spectrum of one of the speakers' voices is consistently louder than the other's across the entire analysis frequency range, the system will not be able to detect the pitch of the weaker speaker. Furthermore, even if estimation of the quieter speaker's pitch was still possible in this scenario, separation

---

[5]A correlogram is a two dimensional function displaying the autocorrelation function as a function of filter channel center frequency.

based on pitch differences alone is not always sufficient; Weintraub notes that other features of the filter channels, such as pitch dynamics, degree of periodicity, and changes in amplitude, should be used in addition to pitch.

2.2.4 Weintraub's Computational Model for Separating Two Simultaneous Talkers (1986)

In [43], Weintraub proposes a system for performing separation of simultaneous speech based upon an auditory model of the cochlea. The system builds upon his previous work presented in [42]; the same auditory model as described in section 2.2.3 is used. However, a number of important additions were made, including: 1) an algorithm for tracking the pitch period of each of the two speakers, 2) a method for estimating the voicing characteristics (voiced/unvoiced) of each speaker's voice, 3) a method for estimation of the spectrum of each speaker's voice, and 4) a method for resynthesis of the speech waveforms from these spectral estimates.

The system was evaluated by feeding separated speech, generated by the system, into a separate continuous-digit-recognition system. The processing resulted in an improved recognition rate of 57%, up 13% from the 44% recognition rate of the same recognition system on the unprocessed co-channel speech. While such improvements are encouraging, the results still fall far below the 87% recognition rate achieved on clean speech. Intermediate results presented in the paper indicate that there is room for improvement in each of the processing stages, e.g. pitch estimation, voicing estimation, etc.

2.2.5 Naylor and Boll's extensions of HMS (1987)

In [27], Naylor and Boll propose several enhancements and extensions to the Harmonic Magnitude Suppression (HMS) system described in section 2.2.2. The new system is shown schematically in Fig. 2.4. As with the original system, the focus is on the negative TIR case; i.e. when the interfering speaker is louder than the target speaker. However, unlike the original system, the new system allows for estimation of the interfering speaker's pitch and source-state (voiced, unvoiced, or silent) directly from the

co-channel signal. Furthermore, the new system also provides for attenuation of unvoiced interference, in addition to voiced interference. First, the authors evaluate four single-pitch estimation methods for determination of the louder, interfering speaker's pitch, and determine that the Maximum Likelihood Pitch Estimation technique [44] offers the best performance across a variety of degradations (additive noise, channel effects, co-channel interference).
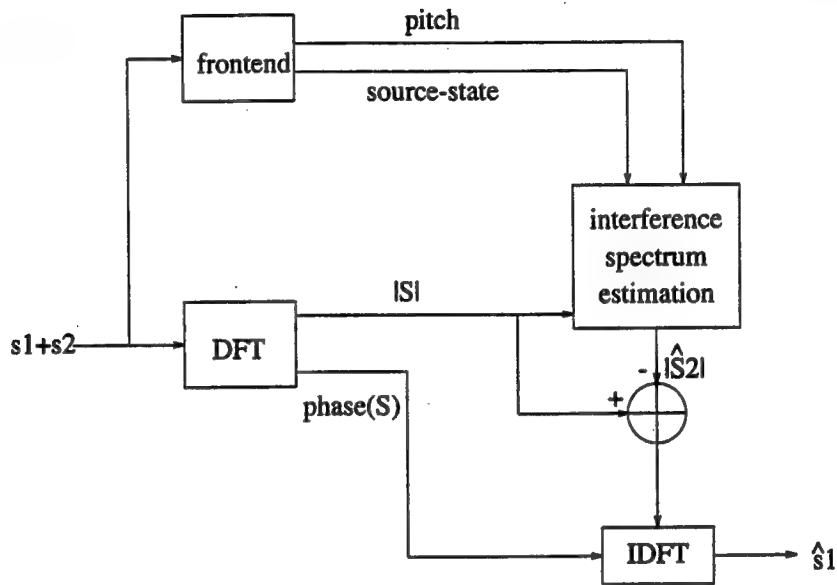
Figure 2.4: Schematic diagram of modified HMS speech enhancement system.

A number of algorithms were tested for determining the source-state of the louder talker. These included: ratio of power at harmonics to total power, error between HMS model spectrum and real spectrum, and the peak correlation value of the Maximum Likelihood Pitch Estimator. The latter was found to perform the best, and was incorporated into the final system. As indicated earlier, estimation of the source-state of each speaker is necessary so that the appropriate type of processing can be used. In the case of voiced interference, the standard HMS method of spectral suppression is used, as detailed in section 2.2.2. In the case of silence, no processing is done at all; i.e. the input signal is passed through unprocessed, straight to the output. To handle the suppression of unvoiced interference, three techniques were evaluated: lowpass filtering, inverse spectral filtering, and smoothing by cepstral liftering; lowpass filtering

was found to perform attenuation of unvoiced interference with the least attenuation of the (voiced) target signal.

The system offers some important improvements over the original HMS system, including estimation of the interference pitch from the co-channel signal, estimation of the louder talker's source-state, and suppression of unvoiced interference. Nevertheless, it still has several limitations. First of all, it still relies on the assumption that the interfering speaker is the louder of the two speakers; such a limitation isn't very practical since, if this assumption is violated, the system will attenuate the target speaker whenever his/her voice becomes louder than the interfering speaker's voice. Such a limitation might be circumvented if the system provided a method for maintaining continuity of speaker identity from frame to frame; however the system does not. Furthermore, even if the target speaker's voice is always quieter than the interfering speaker's, the pitch estimator will not always pick the pitch of the louder, interfering speaker, especially if the power of the target speaker's voice isn't considerably lower than that of the interfering speaker, or if the target speaker has a strong first formant relative to the interference speaker. Another limitation of the system is it's inability to estimate the pitch and voicing of the quieter, target speaker; the target speaker's voice is estimated as the residual left after suppression of the interference speaker's voice. As HMS will not fully cancel the voice of the interfering speaker, it would be desirable to be able to enhance the estimate of the target speaker's voice, such as by comb-filtering, in the case where the target speaker's voice is voiced. However, such enhancement requires knowledge of the target speaker's pitch, and the decision whether or not to perform such enhancement depends on knowledge of the target speaker's source-state.

2.2.6   Lee and Childers' Co-channel Speech Separation via Multisignal MCESA (1988)

In [16], Lee and Childers propose a method for separation of co-channel speech signals based on the multisignal Minimum Cross Entropy Spectral Analysis (MCESA) method. Multisignal MCESA [12] is a method for estimating the power spectrum of one or more independent signals from a co-channel signal when prior estimates of each

spectra are available. Lee and Childers do not propose a new method for generating the initial power spectra estimates, but instead utilize Hanson and Wong's HMS method [10] described in section 2.2.2, which is well suited for the negative TIR cases they address. They note that Parsons' Method of Harmonic Selection [30], as described in section 2.2.1, is better suited for situations where the target and interference power are roughly equal, i.e. TIR $\approx$ 0 dB. Once initial estimates of the target and interference power spectra are constructed via the HMS method, their inverse STFTs are computed to obtain the corresponding autocorrelation functions. These autocorrelation functions and the autocorrelation function of the co-channel signal are then fed to the multisignal MCESA processor. The multisignal MCESA processor generates new estimates of the target and interference power spectra which are consistent with the characteristics of the autocorrelation function of the co-channel signal. A block diagram of the system is shown in Fig. 2.5
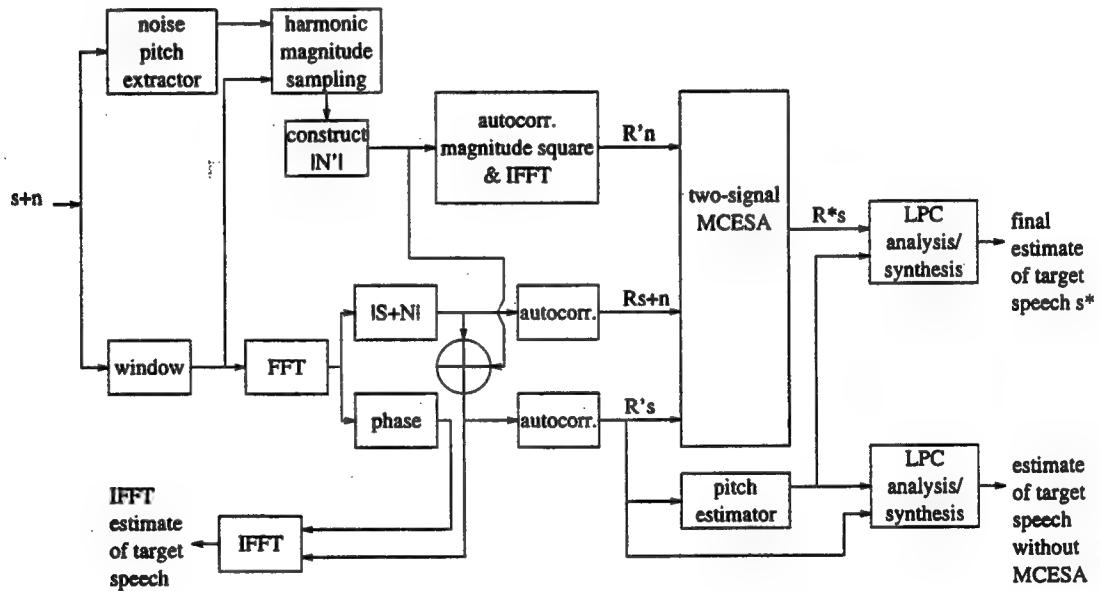
Figure 2.5: Schematic diagram of multisignal MCESA speech enhancement system.

Lee and Childers reported quantitative results showing that the multisignal MCESA method produces enhanced speech which has lower spectral distortion with respect to the original speech signals than does the HMS method alone. Similar results were posted for subjective measures obtained from human listeners. However, the proposed

A-24

multisignal MCESA method is not a standalone speech separation algorithm, but instead is a method for *improving* initial estimates of separated speech spectra generated via another separation algorithm, in this case, Hanson and Wong's HMS method or Parsons' harmonic selection method. As such, it is subject to the same limitations of these methods, which are discussed in section 2.2.2 and section 2.2.1, respectively.

2.2.7 Min et al.'s Automated Two Speaker Separation System (1988)

In [25], Min et al. propose a method for co-channel speech separation based upon the use of variable frame size orthogonal transforms. The processing proceeds along two parallel paths, "channel A" for the first speaker, and "channel B" for the second. A diagram of the system is given in Fig. 2.6. For each channel, the following steps are performed: First, the pitch of the target speaker of the given channel is estimated by picking the maximum in the the standard short-time ACF (autocorrelation function)

$$R_n(k) = \sum_{m=0}^{N-1-k} x(n+m)x(n+m+k) \tag{2.1}$$

within the range of lag values k corresponding to feasible human pitch periods, and if this fails to yield an obvious pitch estimate, the AMDF (average magnitude difference function):

$$AMDF(n,k) = \sum_{m=0}^{N-1} |x(n+k) - x(n+m+k)|_{,,} \tag{2.2}$$

or the ACF of the modified speech signal, $x^5(n)$, is computed to generate a pitch estimate. At each frame, estimates of the given channel's target speaker's pitch are guided by estimates from the previous frame's and the next frame's pitch estimates. Once the pitch has been estimated, an orthogonal transform such as the FFT (Fast Fourier Transform) or FTT (Fast Triangular Transform) is computed over a frame with length equal to an integral multiple $n = 4$ of the pitch period. Next, the transformed speech is estimated to be either voiced or unvoiced/silence by comparing the ratio of the energy of the transform coefficients at harmonic frequencies (indices) to the energy of all the transform coefficients. Again, estimates of voicing parameters from the previous and next frame are used to guide the current frame's estimate. If the current frame is estimated to be voiced, the corresponding time domain signal is synthesized by setting

A-25

the non-harmonic transform coefficients to zero and computing the inverse transform of the resulting modified "spectrum" of transform coefficients. However, if the frame is estimated to be unvoiced or silence, the corresponding time domain signal is estimated by subtracting the other channel's (voiced) waveform estimate from the co-channel signal.

channel A: <A> + B

```
          ┌──────────┐   ┌──────────┐   ┌──────────┐   ┌──────────┐   ┌──────────────┐
          │   FFT    │   │ speech A │   │spectrum A│   │   IFFT   │   │post-processing│
   A ───▶ │   s_A    │──▶│ detection│──▶│estimation│──▶│   s_A    │──▶│  filtering   │──▶ A
          └──────────┘   └──────────┘   └──────────┘   └──────────┘   └──────────────┘
multi-
step
pitch
detect.
   B ───▶ ┌──────────┐   ┌──────────┐   ┌──────────┐   ┌──────────┐   ┌──────────────┐
          │   FFT    │   │ speech B │   │spectrum B│   │   IFFT   │   │post-processing│
          │   s_B    │──▶│ detection│──▶│estimation│──▶│   s_B    │──▶│  filtering   │──▶ B
          └──────────┘   └──────────┘   └──────────┘   └──────────┘   └──────────────┘
```

channel B: <B> + A

< >: target signal
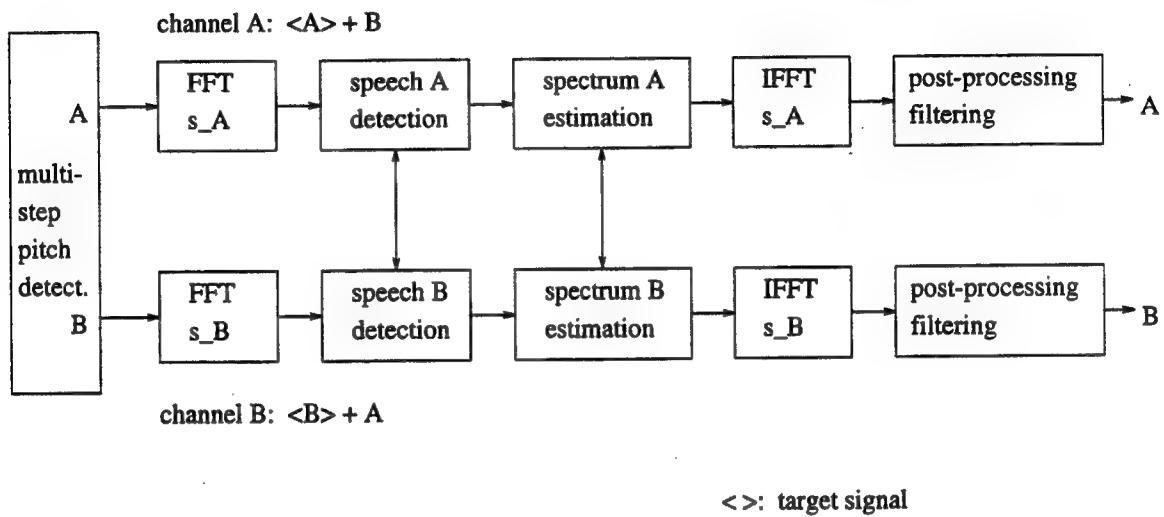
Figure 2.6: Schematic diagram of automated speaker separation system of Min et al.

The system was evaluated by subjective listening tests. Listeners were asked to identify words or phrases of speech extracted from a co-channel speech signal. The increase in intelligibility of the processed (separated) co-channel signals over the unprocessed signals was reported to be "very high." The proposed system includes a number of important features which are not addressed by many of the other proposed methods: Provisions are made for determination of each speaker's voicing, i.e. voiced or unvoiced, and a method for maintaining continuity of speaker identity across processing frames is also provided. Nevertheless, the system still exhibits a number of limitations. First of all, the system will not perform well when the power of the two speakers is not roughly equal. In this case, estimation of the weaker speaker's pitch will be susceptible to error, since its peak in the autocorrelation function will possibly be confused among the other, spurious low-level peaks, and the peaks of the other speaker's harmonics. Another limitation is the assumption of pitch stationarity within the analysis frame of 4 periods.

As discussed in section 2.2.1, the pitch of a speech signal can change by up to 10% from period to period, and the resulting pitch stationarity causes broadening of harmonic peaks in the frequency domain. Because of this assumption of pitch stationarity over the analysis frame, only those bins in the FFT which correspond to harmonic multiples of the estimated pitch are preserved when the separated speech is being resynthesized; the remaining bins are set equal to zero and the resulting modified FFT spectrum is inverse transformed to generate the time domain waveform. Since the harmonic peaks will be spread across a number of bins surrounding each actual harmonic bin, simply sampling the FFT at the harmonic bins will introduce distortions, especially at higher frequencies, where the harmonics will span a larger number of bins around each harmonic bin. Even if the pitch is stationary, such sampling of the FFT will still lead to distortions of the recovered signal, as it ignores the spectral lobe width around each harmonic due to the finite length analysis window. Another limitation is the assumption of pitch periods which are integral multiples of the sampling period. Since it is rather unlikely that the true pitch periods will be integral multiples of the sampling period, the harmonics will not be exactly aligned with the bins of the FFT (or FTT). This will introduce some distortion in the time domain signal when the FFT containing only "harmonic" bins is inverse transformed, even if the pitch is stationary within the analysis frame. Finally, the simple autocorrelation method of pitch estimation suggested here will not be able to accurately estimate the pitch of both speakers when their pitch periods are within a few sampling periods of each other. In such situations, the discrete nature of the autocorrelation function and the width of each speaker's peak in the autocorrelation function will effectively limit the minimum frequency resolution to several multiples of the sampling period.

2.2.8 Quatieri and Danisewicz's Method of Co-channel Interference Suppression Using a Sinusoidal Speech Model (1990)

In [31], Quatieri and Danisewicz propose a method for suppressing interfering co-channel speech by utilizing a sinusoidal model for the speech signals. In the method, which assumes both voiced target and interference speech, the speech signals are modeled over

short time frames as sums of sinusoids of various frequencies, amplitudes, and phases. The authors investigate the two cases: 1) when *a priori* knowledge of the harmonic frequencies of each speaker is available, and 2) when no such *a priori* information is available, and the fundamental frequencies, and thus harmonic frequencies, must be estimated directly from the co-channel signal. Once the harmonic frequencies are obtained, the amplitudes and phases of the corresponding sinusoids are simultaneously estimated via a least-squares approach which operates on the STFT of the frame of co-channel speech. In certain cases, the least-squares solution becomes ill-conditioned when some harmonic frequencies of the two speakers become too closely spaced. In this case, the parameters of those harmonics are linearly interpolated from the parameters of adjacent frames. Finally, enhancement of the target speech signal is achieved by synthesizing the waveform consisting of only those sinusoidal components attributed to that speaker.

Pitch estimation is accomplished by first initializing the two speakers' pitch contours with pitch estimates obtained by a finite-grid gradient search minimization of the difference (error) energy between the actual co-channel frame and the synthesized co-channel frame. The boundaries of the grid are determined by the minimum and maximum allowable pitch frequencies. Once these initial estimates are obtained, pitch estimates for successive frames are obtained by minimization of the same cost function, but over a frequency region restricted to a local neighborhood of the previous frame's pitch estimates. The entire system is illustrated in Fig. 2.7.

The method reportedly offers effective enhancement of the target speaker from TIRs of 9dB to -16dB when *a priori* knowledge of the two speakers' pitches is available. However, when the pitches are estimated directly from the co-channel signal, enhancement is only achieved when the two speakers' power are roughly equal; i.e. 0dB target-to-interference ratio. The method shares several of the limitations of many previous approaches: reliance on the assumptions of only voiced target and interference speech, and the assumption of continuous pitch tracks (i.e. no pauses of silence). Furthermore, unlike Parsons' method described in section 2.2.1, pitch contours are assumed to be non-intersecting. There are no provisions for pitch tracking, but rather frame-to-frame
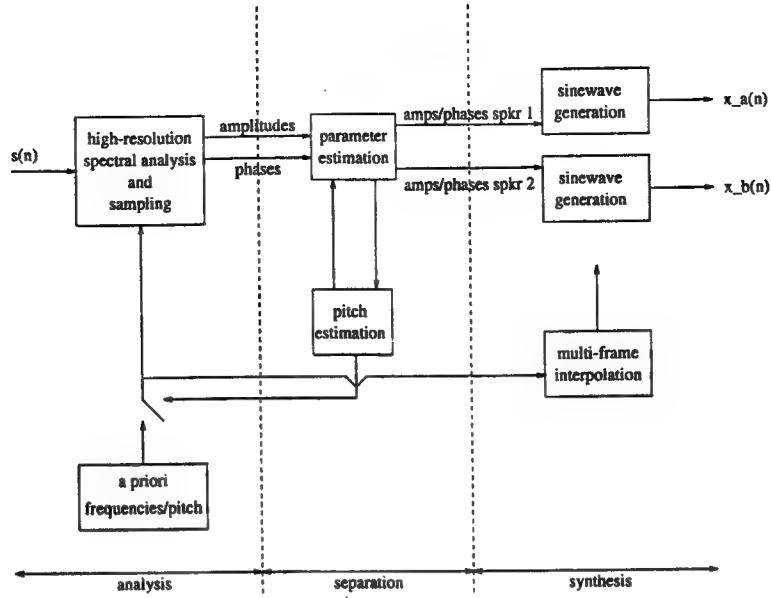
Figure 2.7: Schematic diagram of Sinusoidal Modeling Co-Channel Interference Suppression System.

consistency of speaker identity is maintained by always assigning the lower pitch estimate at each frame to the same speaker and the higher pitch estimate to the other speaker. Such an approach is quite fragile in that if one of the pitch estimates is spurious, *both* of the resulting pitch assignments may be incorrect. Furthermore, since each frame's pitch estimates are initialized with the pitch estimates of the previous frame, any spurious pitch estimate will lead to erroneous pitch estimates for all successive frames.

Another limitation is the assumption of pitch stationarity within the analysis frames. As discussed in section 2.2.1, such non-stationarity results in increased broadening of harmonic peaks with increasing frequency. The assumption of pitch stationarity in the model, as reflected by the modeling of the speech as consisting of purely sinusoidal components, implies ideal harmonic shapes in the STFT determined only by the shape of the time window used to weight the frame. Since, for true speech, the harmonic components aren't truly sinusoidal, the shape of the true pitch harmonics will deviate from the ideal peak shapes assumed in the model. Thus, the estimates of the harmonic amplitudes and phases obtained via the employed least-squares method will be subject

A-29

to error, even if the estimates of the harmonic frequencies are accurate. This will result in distortions of the reconstructed speech signal.

2.2.9 Assman and Summerfield's Modeling of the Perception of Concurrent Vowels (1990)

In [1], Assman and Summerfield investigate two computational models as potential mechanisms for modeling the perception of co-channel voiced speech. Each model is based upon an auditory model decomposition, in which the input signal is passed into a parallel filter bank of 256 roughly non-overlapping bandpass filters, which span the 0–6.2 kHz frequency range. The output of each channel is then passed through a compressive non-linearity, which acts as an automatic gain control (AGC), effectively reducing the dynamic range of the channel output. In the first model, called the "place" model, the representation of a given input sound is given by the relative distribution of power levels across the filter channels, whose center frequencies roughly correspond to different "places" along the cochlea. The resulting representation is analogous to a magnitude Fourier spectrum. A schematic of the place model is shown in Fig. 2.8. For this model, determination of the two speakers' pitches was accomplished via the Modified Duifhuis-Willems-Sluyter (MDWS) method. This method employs a series of "harmonic sieves" which sample a signal's magnitude spectrum at harmonic frequencies of a given fundamental frequency. The goodness of fit of a fundamental frequency estimate is given by the number of spectral harmonics which are passed by the corresponding harmonic sieve, and the fundamental frequency whose harmonic sieve best fits the spectrum is chosen as the pitch. For the estimation of *two* pitches, the fundamental frequency *pair* is chosen as that which jointly maximizes the number of spectral harmonics passed by the two corresponding harmonic sieves. The magnitude of the individual speaker's spectra are estimated by sampling the place model "spectrum" of the co-channel speech at the harmonic frequencies of each estimated pitch.

In the second model, the "place-time" model, the standard short-time autocorrelation function (ACF), as given by Eq. 2.1, is computed for each filter channel's output. The ACFs for all channels are then summed to form a pooled ACF, and the two highest
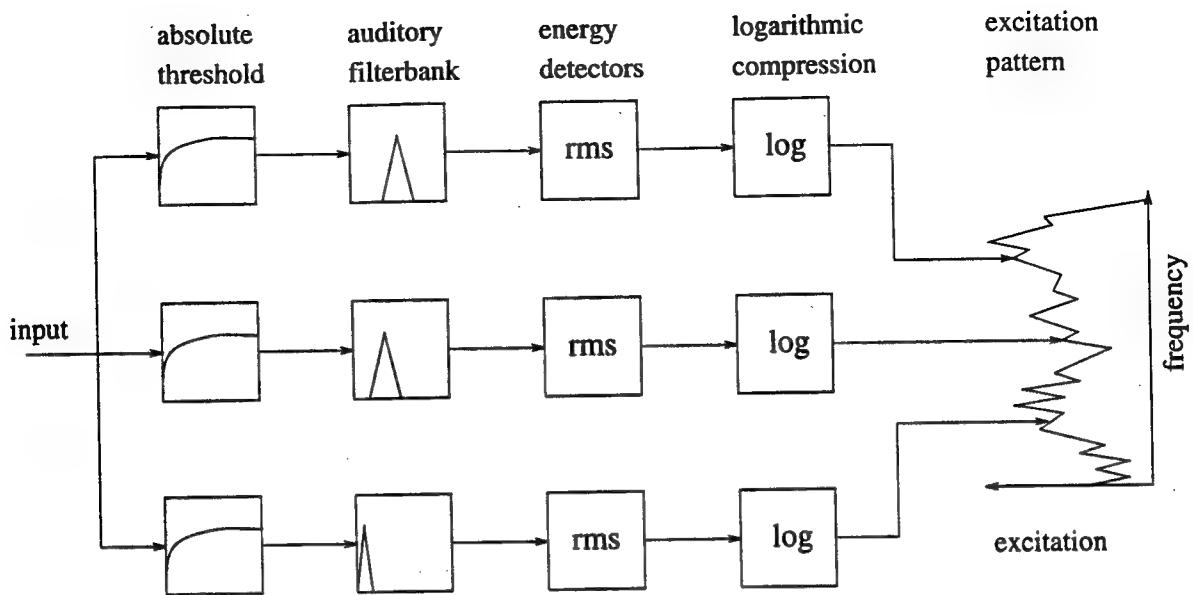
Figure 2.8: Schematic diagram of Assman and Summerfield's place model.

peaks in this ACF are taken to be the pitch periods of the two speakers. The magnitude of the individual speaker's spectra at the channel center frequencies are then estimated by sampling the individual channels' ACFs at time lags corresponding to each of the two estimated pitch frequencies. The model is shown in Fig. 2.9.

The two models were evaluated first on pitch estimation alone, and then on a vowel identification task. In the pitch estimation tests, the place model performed relatively poorly, with a mean pitch estimate error of approximately 7 Hz over 300 test utterances. The poor performance was attributed to inadequate frequency resolution of the model which, in turn, depended on the frequency resolution of the constituent bandpass filters. The place-time model was found to perform significantly better than the place model on the same task, averaging a mean pitch estimate error of approximately 3 Hz over the same test data. The improved performance over the place models is due to the reliance of the place-time model upon resolution in the time domain, rather than the frequency domain, for identification of periodicities. In the vowel identification task, two vowels were summed to form a composite vowel. The composite vowels were then subjected to the two separation models, and the resulting two separated sounds were then input to a template-matching classifier. The place-time model was again found to

A-31
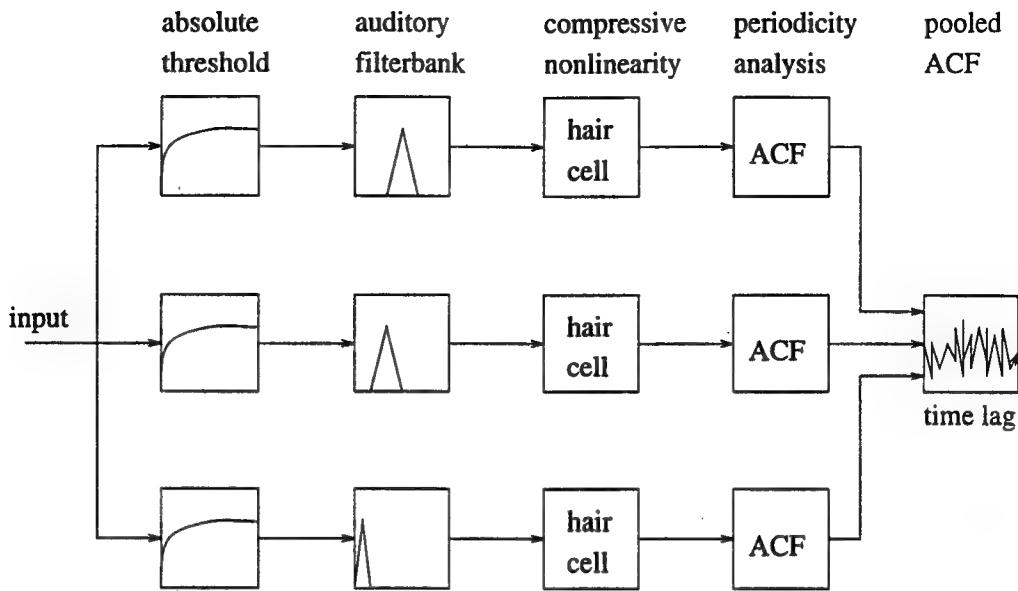
Figure 2.9: Schematic diagram of Assman and Summerfield's place-time model.

perform better than the place model for both pitch estimation of each of the constituent vowels and estimation of the individual vowels' spectra. Absolute performance was somewhat poorer than the average performance of human subjects presented with the same unprocessed composite vowels.

The methods presented in this paper demonstrated some degree of speech separation in the highly constrained task of vowel separation. However, they suffer from a number of significant limitations which limit their utility in a practical system. As stated before, the method is designed only for voiced on voiced speech. Also, the approaches are offered only as methods for performing the actual *separation* of frames of co-channel speech; as such, there are no provisions for estimation of the number of speakers present or for re-assembling the separated frames into continuous speech. Another, perhaps more significant, limitation is that no means are provided for resynthesizing the separated speech signals from the separated spectra. In each of the two models, the magnitude of the separated spectra is only evaluated at the center frequencies of the channel bandpass filters, not at the harmonic frequencies of the associated fundamental frequency. While reconstruction of the time domain signals might be accomplished by interpolation of the magnitude spectrum at the harmonic frequencies and inverse Fourier transforming

of the resulting zero-phase spectrum, the wide spacing of the channel center frequencies at lower frequencies may be insufficient to accurately sample the formant structure of the true spectrum. This limitation is significant, regardless of whether the intended final consumer of the separated signals is a human or a machine, as distortions in the spectral representation of a given sound's formant structure may result in incorrect interpretation of the reconstructed sound. Another limitation of the models is the requirement of approximately equal power of the two speakers (vowels). In the event that one speaker is significantly louder, even the better performing place-time model will suffer performance degradation in estimation of the quieter speaker's pitch, since the model implicitly assumes that the second highest peak in the pooled ACF represents the pitch of the weaker speaker, and not a harmonic or sub-harmonic of the louder ·speaker.

2.2.10 Naylor and Porter's Speech Separation System (1991)

In [26], Naylor and Porter propose a method for separation of voiced speech by use of AR (autoregressive) spectral estimates and spectral subtraction in the complex domain. First the co-channel speech is divided into 40 msec frames. Next, a high-order AR spectral estimate of the frame's magnitude spectrum below 550 Hz is generated using the modified covariance method. The use of such a parametric spectral estimator allows for resolution of harmonic peaks, even for the short analysis durations, and for detection of the quieter speaker's harmonic peaks, which would be undetectable in a periodogram of the same analysis frame. With the peaks identified in this AR spectral estimate, estimates of both speakers' pitch are generated by clustering the peak frequencies into two groups, whose constituents could be harmonically related to a feasible human pitch value. Once the two pitch estimates are generated in this way, an estimate of the complex (as opposed to magnitude-only) spectrum of one speaker is constructed as a linear combination of scaled complex-valued peaks centered at integral multiples (harmonics) of that speaker's pitch estimate and with amplitudes obtained by sampling the Fourier spectrum of the co-channel signal at those frequencies. Unlike Hanson and Wong's HMS method described in section 2.2.2, and related methods which

assume a peak shape which is simply a scaled and shifted version of the magnitude inverse Fourier transform of the of time window used (e.g. Hamming, Hanning, etc.), the method presented here describes the peak shape as a complex-valued, non-linear function of the time window. Presumably, this allows for modeling of the effects of the non-stationarity of speech signals. In areas where harmonic peaks overlap, a least-squares method, analogous to that of Quatieri and Danisewicz described in section 2.2.8, is used to estimate the overlapping harmonic peaks' amplitudes and phases. Finally, separation is accomplished by subtraction of the complex-valued spectrum estimate from the complex spectrum of the co-channel signal, leaving the complex spectrum of the other speaker's voice. A diagram of the system is shown in Fig. 2.10.
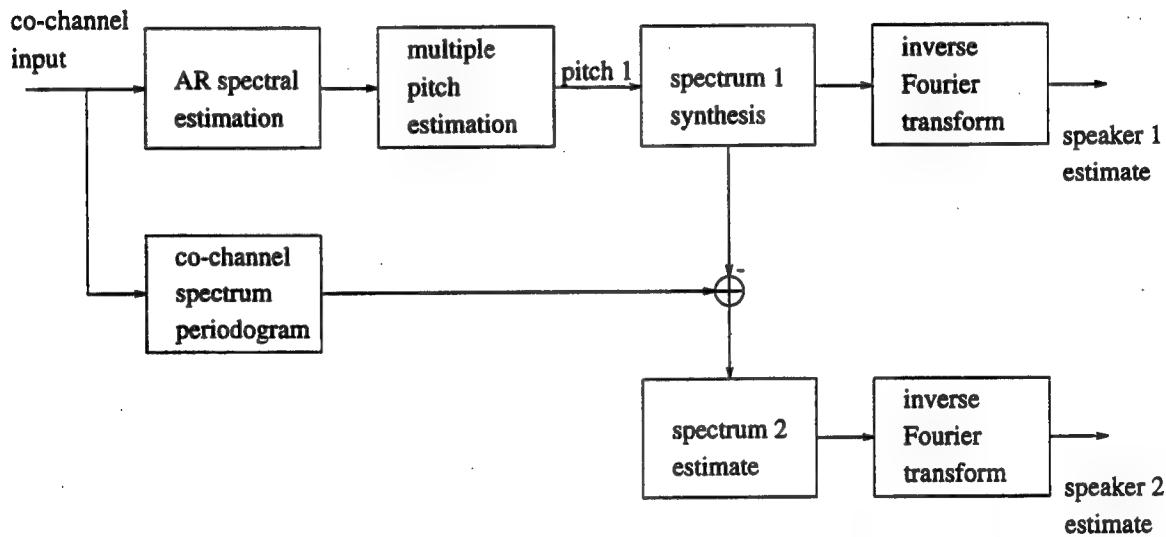
Figure 2.10: Schematic diagram of Naylor and Porter's speech separation system.

The proposed system was reported to offer improved word recognition rates on co-channel signals at a TIR of -14 dB. Furthermore, it addresses an important problem which is not dealt with by many of the other methods, namely the assumption of pitch stationarity within the analysis interval. Thus, the method is apt to introduce less spectral error than methods which assume constant pitch, and thus, ideal harmonic peak shapes. Nevertheless, the system still faces a number of limitations in common with most other systems. First is its reliance upon the assumption of voiced on voiced

speech; no accommodations are made for the processing of voiced/unvoiced combinations. Second, the related tasks of detection of the voicing of each speaker (i.e. voiced or unvoiced) and determination of the number of speakers actually speaking at a given instant, are not addressed. Finally, no provisions are made for assigning the frames of separated speech to one or the other speaker, so as to assemble the individual separated frames into continuous utterances. In addition to these, the proposed methods faces a number of limitations particular to itself. First is the sensitivity of AR spectral estimates in low SNRs; it is well known that the AR estimates of pole locations are not very reliable when the SNR is not high. Since the pitch estimates of this method rely upon accurate estimation of the pole locations corresponding to harmonic peaks, the pitch estimates will not be very reliable in low SNR situations. Without accurate pitch estimates, the successive spectral subtraction will not sample the complex co-channel speech spectrum at the correct frequencies, resulting in poor separation. Due to these limitations, the practical utility of the system in a realistic setting is diminished.

2.2.11 de Cheveigné's separation of concurrent harmonic sounds using a time-domain cancellation model of auditory processing (1993)

In [6], de Cheveigné presents a method for separation of two harmonic sounds based upon time-domain multi-notch filtering of the co-channel signal. First, estimates of the two speakers' pitches are generated using a modification of the AMDF (average magnitude difference function) called the DDF (double difference function). Whereas in the conventional AMDF, the single pitch period estimate at time instant n is given by the lag index k which minimizes the AMDF function (Eq. 2.2 repeated here):

$$\text{AMDF}(n, k) = \sum_{m=0}^{N-1} |s(n+k) - s(n+m+k)|, \qquad (2.3)$$

for the speech signal $s(n)$, in the DDF, *two* pitch estimates are obtained by selecting the *pair* of lag indices $\{k, l\}$ which minimizes the DDF:

$$\text{DDF}(n, k, l) = \sum_{m=0}^{N-1} |s(n+m) - s(n+m+k) - s(n+m+l) + s(n+m+k+l)|$$

$$(2.4)$$

over a two-dimensional $\{k, l\}$ grid spanning the range of pitch periods corresponding to the range of feasible human pitches. The DDF corresponds to the output of the cascade of two multi-notch filters whose fundamental periods are given by the lags k and l. If the two pitch estimates are accurate, each notch filter should null out one of the voices, and the output of the cascade should ideally attain its minimum value, namely zero. de Cheveigné does not recommend a new method for reconstruction of the two speech signals once the two pitch estimates have been obtained, but does, however, review the strengths and weaknesses of the commonly used methods of enhancement of voiced speech by comb-filtering and attenuation of interfering voiced speech by use of a multi-notch filter.

The algorithm was evaluated on a pitch estimation task in which two pitch estimates were generated for each of a number of pre-selected frames of co-channel speech which were determined to be "clean," or strongly periodic. 90% of the pitch estimates fell within 3% of an octave from the actual pitch, or a harmonic or sub-harmonic of that pitch. de Cheveigné specifically states that he "does not attempt to design a complete system for speech separation." Rather, the intent was merely to devise a method for estimation of the pitches of co-channel speech signals and to propose a physiological basis by which the time-domain multi-notch-filtering might be accomplished. In light of this fact, the lack of key components, such as means for reconstruction of the separated signals, estimation of the number of speakers and their voicing, and re-assembly of the separated speech signals into continuous utterances, is understandable. Nevertheless, evaluating the method strictly as a pitch estimator for co-channel speech, we find that it still faces number of limitations. First of all is the limited frequency resolution afforded by the DDF method. As the DDF is only evaluated at lags $\{k, l\}$ equal to integral multiples of the sampling period, the frequency resolution of the DDF pitch estimates will be quantized accordingly. This issue is not unique to this estimator; many standard pitch estimators are also subject to such granularity resulting from a finite sampling rate. This problem might be alleviated somewhat by increasing the sampling rate or by interpolation of the DDF, but since the optimal $\{k, l\}$ pitch estimate pair is found by an *exhaustive* search of the k–l grid, doing so will increase computation time four-fold

per every doubling of resolution. Additionally, the AMDF estimator also assumes pitch stationarity within the analysis interval. The analysis frame must be long enough to accommodate at least four cycles of the lowest frequency pitch. However, this results in an excessive window length for higher pitches. In the case of a signal steadily increasing or decreasing in frequency from the start of the analysis frame to the end, the AMDF will generate a pitch estimate corresponding to the pitch of the signal near the start of the frame. Such a pitch estimate is not appropriate for comb- or notch-filtering of the entire frame. Most pitch estimators which operate on fixed frame lengths also suffer from the problem of requiring frame lengths which are long enough to include at least two cycles of the lowest frequency expected, resulting in an excessive frame length for the higher frequencies. However, in the case of the DDF, this problem is exacerbated by the fact that the frame length must be long enough to include two cycles of two signals with the lowest expected frequency, resulting in a minimum frame length of *four* times the longest expected pitch period.

2.2.12 Chazan et al.'s Optimal Multi-Pitch Estimation Using the EM Algorithm for Co-Channel Speech Separation (1993)

In [5], Chazan et. al present a method for performing separation of voiced speech which utilizes the EM (Estimate-Maximize) algorithm to optimally estimate (in the Maximum Likelihood sense) the pitch of each speaker. In addition, the pitch of each speaker is allowed to vary linearly within the analysis frame. First, each speech signal is modeled over the duration of a frame as a quasi-periodic signal; i.e. a periodic signal which has been time warped by a warping function $\phi(t)$:

$$ s(t) = \sum_{k=1}^{L(\phi(t))} c_k e^{jk\phi(t)}, \quad t \in (-\frac{T}{2}, \frac{T}{2}). $$

The warping function $\phi(t)$ is constrained to be of the form:

$$ \phi(t) = \frac{1}{2}\alpha t^2 + 2\pi\beta t $$

where β is the average pitch over the frame and α is the rate of change of pitch over the frame, so that the instantaneous frequency, given by $\dot{\phi}(t)$, is linear in t. It is shown

that determining the Maximum Likelihood estimate of each speaker's pitch and rate of change of pitch is equivalent to finding the warping function $\phi(t)$, parameterized by α and β, which maximizes the output of the cascade of a time-warp given by $\phi(t)$, a comb-filter tuned to 1 rad/sec, and an inverse time-warp given by $\phi(t)^{-1}$. The Multi Pitch Detection Algorithm (MPDA) proceeds as follows: 1) The pitch β and rate of change of pitch α for one of the speakers are found by a numerical maximization of the described likelihood function. The signal resulting from this maximization is then subtracted from the input co-channel signal, leaving an estimate of the second speech signal in the residual. 2) The procedure is repeated again on this residual signal to yield estimates of the second speaker's pitch and rate of change of pitch. The signal resulting from this maximization is then subtracted from the input co-channel signal, producing an estimate of the first speech signal. This estimate is then fed back to step 1), and the procedure repeats, iterating back and forth, using current estimates of α and β to decompose the input signal at each step, resulting in improved estimates of the two speech signals at each iteration, until the procedure converges. The procedure is illustrated schematically in Fig. 2.11.
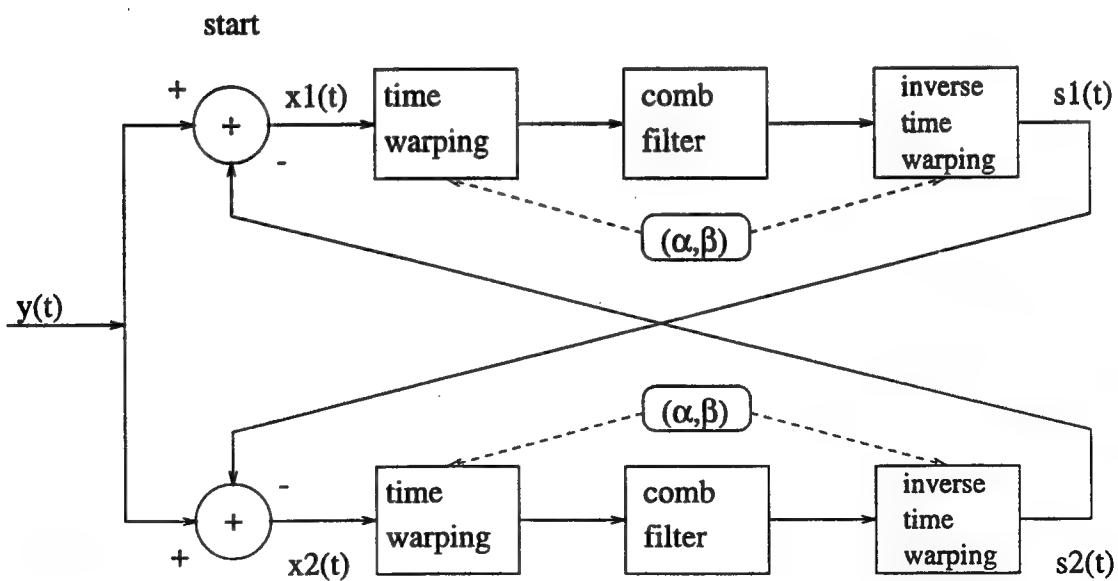
Figure 2.11: Schematic diagram of Chazan et al.'s Multiple Pitch Detection Algorithm.

The MPDA algorithm was tested and compared against three standard pitch estimation algorithms (which had been modified to produce two pitch estimates) on a pitch estimation task on co-channel speech at 0 dB TIR and -12 dB TIR. The MPDA was shown to exhibit smaller pitch errors than the other estimators, at both TIRs. No mention was made, subjectively or otherwise, of the quality of the separated speech signals.

The MPDA algorithm addresses the important issue of pitch non-stationarity within analysis frames. As it allows for a linearly-changing pitch within the analysis interval, it should provide for more effective separation than that achievable by methods which assume constant pitch, for reasons described previously regarding the smearing, or FM broadening, of harmonic peaks due to pitch non-stationarity. However, it still suffers from a number of limitations in common with the other proposed methods for speaker separation. First is the inability to handle cases other than voiced speech on voiced speech; it is implicitly assumed that both talkers' speech is voiced. Second, there are no provisions for detection of each speaker's voicing. Thirdly, there is no mechanism provided for maintaining speaker continuity across the separated frames. The lack of these features limits the practical utility of the proposed method as a complete speaker separation *system*.

2.2.13 Savic et al.'s Co-channel Speaker Separation Based on Maximum-Likelihood Deconvolution (1994)

In [33], Savic et al. propose a method for co-channel speaker separation based upon Maximum-Likelihood deconvolution of the co-channel signal, so as to generate estimates of the excitation signal of each speaker. The resulting excitation signal estimates, when run through the filters of the corresponding speaker's vocal tract models, will produce the restored, separated speech signals. In the algorithm, each speaker's excitation signal $u_i(t)$ is modeled as a "Bernoulli-Gaussian Backscatter sequence," which models the glottal pulse train and fricative noise. Mathematically, the excitation of the ith speaker is modeled as:

$$u_i(t) = r_i(t)q_i(t) + f_i(t)$$

where $q_i(t)$ is a Bernoulli sequence, $r_i(t)$ is a Gaussian scale factor, and $f_i(t)$ is a fricative noise process. Furthermore, it is assumed that the vocal tract response of each speaker can be parameterized by an LPC filter, with impulse response given by $w_i(t)$. Thus, the co-channel signal consisting of M speakers' voices can be represented as:

$$z(t) = n(t) + \sum_{i=1}^{M} w_i(t) * u_i(t)$$

where $n(t)$ represents additive noise. It is then shown that if the vocal tract filter coefficients $w_i(t)$ and the fricative noise variances are known or can be estimated, then the fricative noise sequences and glottal pulse sequences of each speaker can be estimated in a Maximum Likelihood sense. The corresponding speech signals can then be resynthesized by running the estimated excitation sequence of each speaker, which is simply the sum of the estimated glottal pulse and fricative noise sequences, through the corresponding vocal tract filter $w_i(t)$.

The algorithm was informally tested by listening to the speech recovered by the algorithm from co-channel utterances generated by summing two separate single-speaker utterances, and comparing it with the original, individual utterances. Reportedly, the system "performed well." However, a most significant limitation of the algorithm is the assumption that the individual speakers' vocal tract filters are available, or can be estimated from the co-channel signal. In practice, these vocal tract filters are not available a priori, and must be estimated directly from the co-channel signal. Estimation of such filters is tantamount to performing actual separation of the speech signals, and without these filters, the algorithm cannot be used. In other words, in order to be able to estimate the vocal tract filters, or equivalently, the spectral envelope of each speaker's voice, one must first perform some type of speech separation. In that in most realistic scenarios, a priori knowledge of each speaker's vocal tract filter is not available, the practical utility of such an algorithm is limited.

2.3 Multi-Pitch Estimation via an Auditory Model-based cepstral pitch estimator

2.3.1 Background on Auditory Models

Over the past several years, there have been several attempts to incorporate a computational model of the human auditory system into the front-end processing stage of various automatic speech processing systems ([11], [23], [24], [9], [39], [18], [36], [35]). Applications of such models include pitch estimation, as well as spectral representation of sounds for speech recognition. While there are a number of significant differences among the various auditory models, they do share a number of common features. A generic auditory model is shown in Fig. 2.12. The first stage is a large
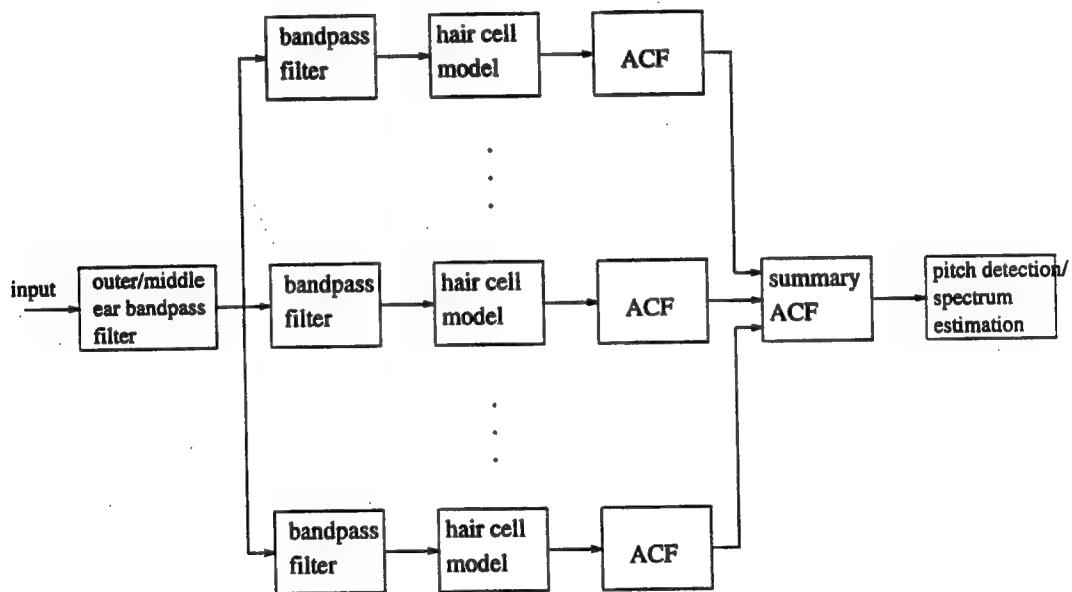
Figure 2.12: Schematic diagram of a generic auditory model.

bandwidth lowpass or bandpass filter, which models the coarse frequency selectivity of the outer/middle ear. Next the signal is fed into a parallel arrangement of processing units, which are commonly referred to as *channels*. Each channel consists of a number of sequential processing stages. The first stage is a narrowband bandpass filter, which models the frequency response at a particular location along the basilar membrane of

A-41

the cochlea. The bandpass filters are often overlapping in frequency, with center frequencies logarithmically spaced from low to high frequencies. The outputs of each filter are then fed into some type of hair cell model, which models the behavior of hair cells in the cochlea, whose function is to transduce the mechanical stimulation of a location along the basilar membrane into nerve firing patterns. Often this stage consists of half-wave rectification and contains an automatic gain control (AGC) of some type as well. Fig. 2.13 shows the channel outputs of a typical auditory model, produced in response to a voiced speech (periodic) signal input. Next, the output of each channel's hair cell
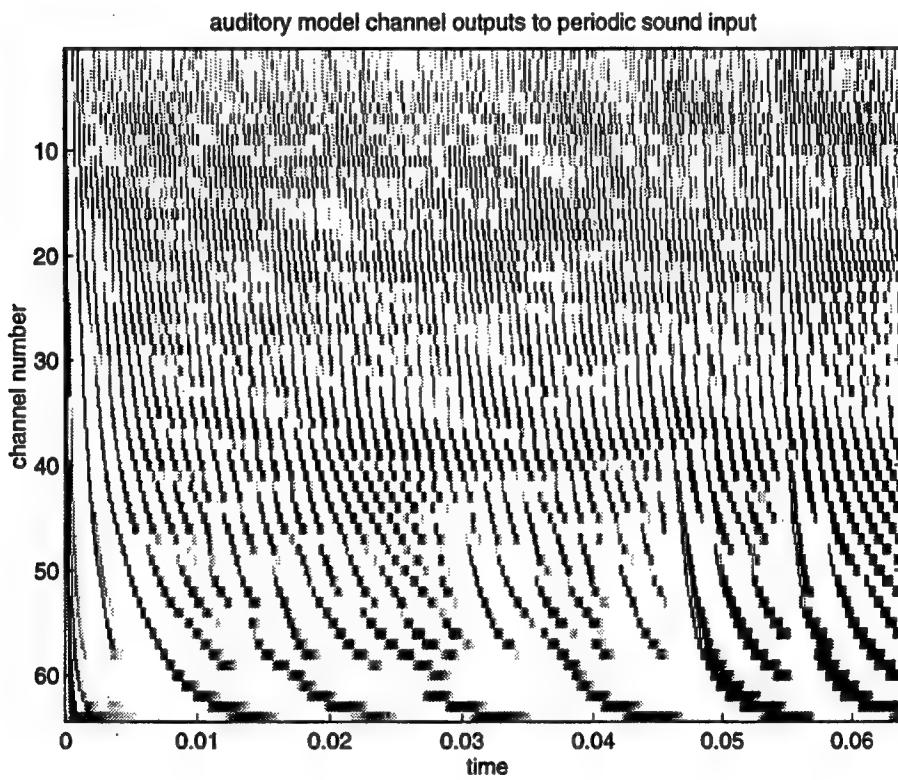
Figure 2.13: Auditory model channel outputs for a periodic sound input.

model is fed into a processing stage which performs some type of periodicity analysis of that channel's output. This stage is almost always implemented as the standard short-time ACF (autocorrelation function), as given by Eq. 2.1, or by a modification thereof. If the individual ACFs are juxtaposed in order of increasing channel center frequency, such that the time lags of each ACF are aligned with the corresponding

time lags of the adjacent ACFs, the resulting representation is known as a *correlogram*. The correlogram shows the spectral energy distribution and time structure of a sound on two independent axes. The use of such a representation for pitch determination is attributed to Licklider, who, in 1951, proposed the "duplex theory" of pitch perception [17]. Finally, the ACF of all channels are summed to produce one "pooled," or "summary" ACF. A periodicity which is present in several channels will be reflected by a peak in the summary ACF, as the corresponding peaks in the individual channels' ACFs will add coherently in the summary ACF to produce a large peak. Conversely, spurious peaks occurring in the individual ACFs will not sum coherently in the summary ACF, and should therefore not produce an significant peak in the summary ACF. Fig. 2.14 shows the correlogram for the same periodic input signal used in Fig. 2.13, and the corresponding summary ACF. Note how peaks (indicated by dark areas) which are consistent across channels add coherently to produce large peaks in the summary ACF, while those peaks which aren't consistent across multiple channels do not produce significant peaks in the summary ACF. The dark vertical line in the correlogram at around time lag 0.009 sec, and the corresponding peak in the summary ACF indicate the period of the input speech signal. In that the short-time ACF is the inverse Fourier transform of the short-time power spectrum of a given signal, the summary ACF also gives an indication of the spectral shape of the input sound signal.[6]

2.3.2   The auditory model cepstral pitch estimator

The auditory model cepstral pitch estimator presented here is based upon Lyon's auditory model as detailed in [37] and [38]. Lyon's model includes most of the processing stages described in the previous section for the generic auditory model. The first stage is a large bandwidth bandpass filter modeling the frequency response of the outer/middle ear. This is followed by a parallel arrangement of narrowband bandpass filters, which

---

[6]Use of the low time-lag summary ACF coefficients is sometimes used to characterize the spectral envelope of a given sound [9], in much the same way that low order cepstral coefficients are often used to represent the spectral shape of speech signals.
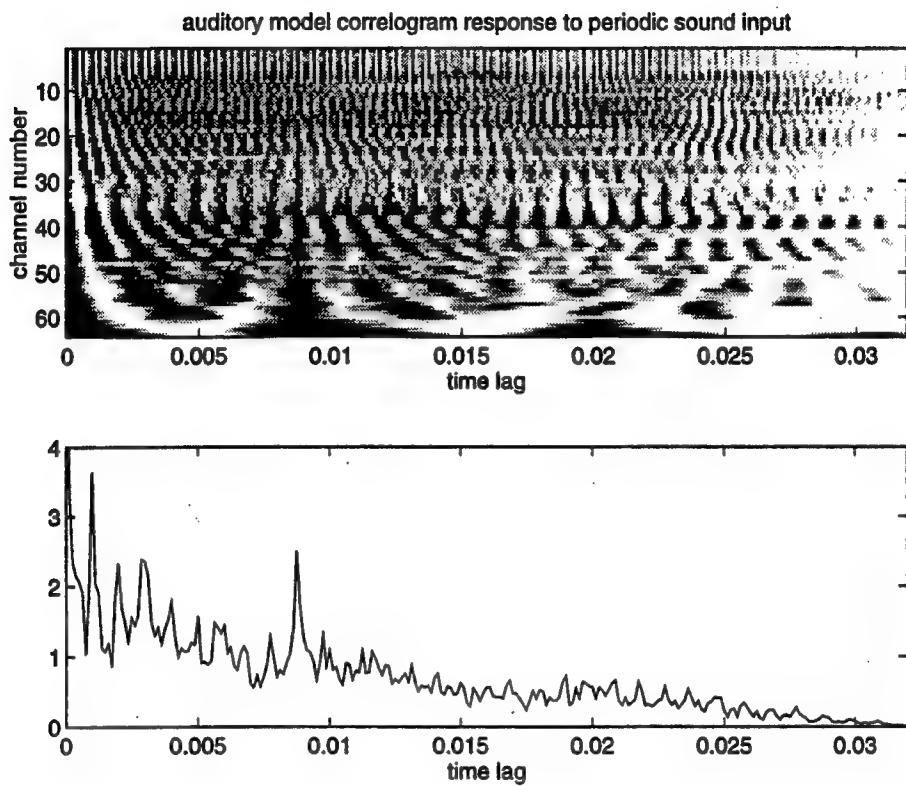
Figure 2.14: Top: Auditory model correlogram for a periodic sound input (darker shades indicate higher amplitude). Bottom: corresponding summary ACF.

collectively model the frequency response at different locations along the basilar membrane of the cochlea, and whose center frequencies are smoothly spaced from low to high frequencies. The output of each bandpass filter is fed into a half-wave rectifier, which models the behavior of the inner hair cells. Then the output of each channel's half-wave rectifier is fed into an automatic gain control (AGC) processing stage, which reduces the dynamic range of the input signal. The short time ACF is computed for each channel's output, and the resulting functions are combined into a correlogram representation. In the pitch detector described in [38] which uses Lyon's auditory model, pitch detection is accomplished by integrating the correlogram across all channels to produce a summary ACF, and picking the maximum peak in this summary ACF within the range of lags corresponding to feasible human pitch periods. Several pre-processing steps are used to enhance the peak structure in the correlogram prior to integration, and some post-processing is done on the summary ACF as well. Nevertheless, the basic pitch estimation procedure is the same as that presented in the previous section for the general auditory model.

The auditory model cepstral pitch estimator is identical to this pitch estimator up to the output of each channel's AGC. Then, rather than computing the short time ACF of each channel output, the (complex) cepstrum is computed. The cepstrum of a signal $x(n)$ is defined as:

$$c(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} \log |X(e^{j\omega})| e^{j\omega n} d\omega$$

This is simply the inverse Fourier transform of the logarithm of the magnitude of the Fourier transform $X(e^{j\omega})$ of the signal $x(n)$. In [29], Noll described the use of the cepstrum for pitch estimation. It was shown that for periodic signals, the cepstrum will exhibit a peak at a time lag equal to the period of the signal, and smaller peaks at multiples of the period, much like the ACF does. This is not surprising, as the ACF is simply the inverse Fourier transform of the power spectrum of a signal; the only difference between the ACF and the cepstrum is the logarithm in the cepstrum and a scale factor difference between the power spectrum $X(e^{j\omega})X^*(e^{j\omega})$ and magnitude spectrum

$|X(e^{j\omega})|$. By computing the cepstrum for each channel, a "cepstrogram" representation, analogous to the correlogram, can be generated. As with the correlogram, the cepstrogram is integrated across all channels to produce a summary cepstrum. Then the time lag of the highest peak in the summary cepstrum, within the range of time lags corresponding to the range of acceptable pitch values, is taken to be indicative of the period of the input signal. Actually, due to the the discrete time nature of the cepstral sequence, the location of the peak is quantized to the nearest integral multiple of the sampling period. To achieve higher resolution, once the peak has been found in the summary cepstrum, a parabola is fitted to that point and the five surrounding data points; the location of the maximum of the fitted parabola is taken to be the refined peak location estimate. Fig. 2.15 shows a cepstrogram generated in response to a periodic sound input, and the resulting summary cepstrum. The dark vertical line in the cepstrogram and the corresponding peak in the summary cepstrum at around time lag 0.0075 sec correspond to the period of the input signal.

To generate pitch estimates for two speakers, first one pitch estimate is generated, as detailed above. Presumably the pitch of the louder speaker will be chosen. Then the frame of co-channel speech is filtered with a multi-notch filter, whose nulls occur at frequencies which are integral multiples of this first pitch estimate. This should remove most of the contribution of the louder speaker's voice from the co-channel signal; the residual signal should ideally contain only the second speaker's voice. This residual signal is then fed back into the auditory model cepstral pitch estimator and the resulting pitch estimate is taken to be the pitch of the second speaker.

The performance of this pitch estimator is evaluated in Chapter 3, where it is compared with several other pitch estimators. However, there are a number of considerations worth mentioning here. First, the method is susceptible to the effects of pitch nonstationarity within the analysis frame. As with the ACF, the window length must be at least twice the period of the lowest allowable frequency. However, such a window length can encompass many cycles of higher frequency inputs with their correspondingly shorter periods. This is not a problem if the pitch is stationary. However, as mentioned previously, the period of speech signals can change by 10% between adjacent cycles. As
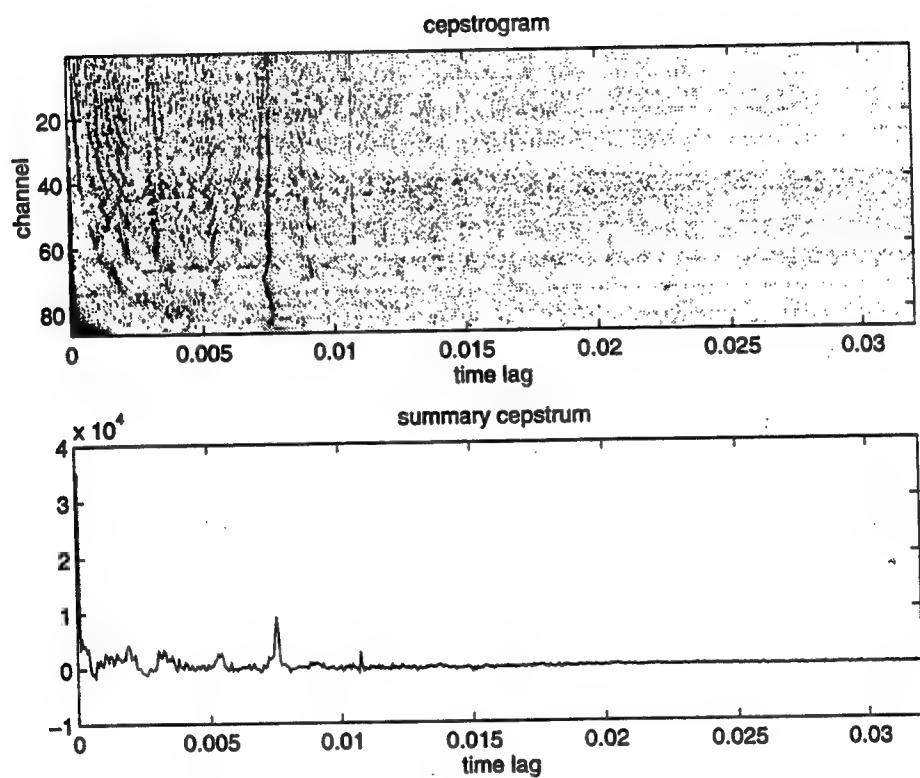
Figure 2.15: Top: Auditory model cepstrogram for a periodic sound input (darker shades indicate higher amplitude). Bottom: corresponding summary cepstrum.

a result, the pitch of a high frequency input signal can undergo significant change within the analysis frame, since as the pitch of the input increases, the number of periods of the waveform within the analysis frame also increases. However, this pitch estimator can only generate a *constant* estimate of the pitch over the frame; it does not provide a measure of the change of pitch over the frame duration. As a result, the amount of separation/suppression attainable with subsequent processing will be reduced. For instance, if a time-domain multi-notch filter was to be used to reduce interference in a frame of co-channel speech in which the interference voice signal had highly-stationary pitch, the filter's effectiveness would be reduced, since its operation depends upon the similarity of the waveform period from cycle to cycle.

Another potential problem may arise if the input signal is extremely bandlimited [32]. In such cases, where there is little periodic oscillation in the log spectrum, there will be no peak in the resulting cepstrum, and as a result, no peak in the summary cepstrum. As a result, any spurious peaks present in the summary cepstrum might be incorrectly identified as that due to the periodicity of the input signal. Fortunately, most speech signals aren't extremely bandlimited.

Chapter 3

Experiments

3.1 Preliminaries

In that most of the proposed methods for co-channel speaker separation rely upon accurate estimation of one or both speakers' pitch, a natural test is to compare the pitch estimation accuracy of the different estimators. Some of the separation methods reviewed do not suggest means for estimation of the constituent speakers' pitches; it is assumed that this information is already available. Works falling into this category include those of Hanson and Wong (section 2.2.2), Lee and Childers (section 2.2.6), and Quatieri and Danisewicz (section 2.2.8). Savic et al.'s work (section 2.2.13) does not use pitch information at all. Of the remaining studies, there is some redundancy with regards to the associated method proposed for performing the actual pitch estimation. This being the case, the pitch estimation methods used in the following studies have been deemed as representative of the different methods presented, and have thus been chosen for testing:[1]

Assman and Summerfield (section 2.2.9) "place-time" auditory model-based pitch estimation.

Chazan et al. (section 2.2.12) EM-based pitch estimation.

de Cheveigné (section 2.2.11) Pitch estimation by the DDF (Dual Difference Function)

Naylor and Boll (section 2.2.5) Maximum Likelihood pitch estimation.

---

[1]Many of the proposed co-channel separation methods employ similar pitch estimation approaches. The choice of this particular subset of studies does not necessarily reflect the originators of a given method, nor the relative superiority of one implementation of a particular method over another. Rather, these have been chosen simply on the basis of being representative of the group.

Naylor and Porter (section 2.2.10) Pitch estimation by clustering of high-order
AR spectrum peaks

The performance of these pitch estimators will be compared with that of the auditory
model cepstral pitch estimator presented in section 2.3. Those of the reviewed studies
still remaining can be accounted for as follows:

Parsons (section 2.2.1) Use of the Schroeder histogram on spectral peak frequencies
is similar to Naylor and Porter's method of clustering peak frequencies.

Weintraub (section 2.2.3) Use of an auditory cochlear model and pooled ACF is
similar to that of Assman and Summerfield's "place-time" auditory model-based
pitch estimator.

Min et al.(section 2.2.7) Use of ACF and AMDF is similar to de Cheveigné's DDF.

Some of the speaker separation methods employ a one-pass approach to estimate
the pitch of both speakers at one time. Others methods are iterative and follow the
general sequence of steps: 1) estimate pitch of speaker 1, 2) use this estimate to suppress
the voice of speaker 1 in the co-channel signal frame, 3) estimate pitch of speaker 2
from the residual signal, 4) (optional) use this pitch estimate to suppress the voice
of speaker 2 from the co-channel frame and feed the resulting residual signal back
to step 1). In this respect, of the methods chosen for testing, those of Assman and
Summerfield, Naylor and Porter, and de Cheveigné can be considered as non-iterative,
while the method of Chazan et al. and the auditory model cepstral pitch estimator
are iterative methods. Note that many of the non-iterative methods can be modified
to be iterative, by following the sequence of steps outlined above. It is assumed that
iterative implementations of the one-pass approaches would yield better pitch estimates.
However, the intent here is to test the pitch estimation algorithms as presented by their
authors, without modification or enhancement. The exception here is the Maximum
Likelihood pitch estimation, as used by Naylor and Boll (section 2.2.5), and presented
in [44]. In Naylor and Boll's work, only the pitch of the louder, interference, speaker
was estimated via the Maximum Likelihood estimator. However, as will be discussed

below, it is often useful to have pitch estimates of *both* speakers' voices. This being the case, the Maximum Likelihood estimator was extended into a dual-pitch estimator by the general 4-step procedure outlined above.

As indicated above, some speaker separation methods only utilize the pitch estimate of one speaker, typically the louder speaker. If the *interference* speaker is assumed to be the louder, then this pitch estimate is used to *suppress* the interference speech, and an estimate of the target speaker's speech is given by the residual signal. If, however, the *target* speaker is assumed to be the louder, then the pitch estimate is used to *enhance* the target speech, with a comb filter, for instance. In the general case, however, it cannot be assumed that the target speaker or the interference speaker is always louder; in some scenarios, both speakers will be at roughly equal power, or the ratio of relative power may change over time. Similarly, it cannot always be assumed that the voice of one speaker can be estimated as the residual left after suppression of the voice of the other speaker in the co-channel signal. This assumption would be violated, for instance, if the co-channel signal contains not only speech signals, but noise, or other interference, as well. In these general cases, it is useful to have pitch estimates of *both* speakers' speech. In this way, the voice of both speakers can be estimated, either directly by signal enhancement techniques, or indirectly, by suppression of the other speaker's voice. Of those studies chosen for evaluation, only that of Naylor and Boll did not present a scheme for estimation of *both* speakers' pitch. Therefore, the Maximum Likelihood *monophonic* pitch estimator used in that study has been modified by the author to generate *two* pitch estimates, so as to be more readily comparable with the other estimators.

3.2 Database

The speech database used for the experiments was recorded at the CAIP Center, at Rutgers University. The database consists of the six sentences listed in Table 3.1. These sentences, taken from [31], have been constructed so as to consist almost entirely of voiced sounds. The entire database was recorded twice, once by a native-American male speaker ("speaker 1") and once by a non-native-American male speaker ("speaker

A-51

A.	"We were away in Walla Walla."
B.	"Our rule will lower your ear away."
C.	"Why were you away a year Roy?"
D.	"All wear your ear low."
E.	"Wear your ear low."
F.	"All rare laws are well."

Table 3.1: Database used in pitch estimation and separation effectiveness experiments.

2"). In addition, an utterance of sentence C was recorded by a third, non-native-American speaker ("speaker 3"). The recording was performed in the relatively quiet, but not anechoic, office environment of the speech processing lab at the CAIP Center. A close-talking microphone was used for the recordings, which were sampled at 16 kHZ and digitized at 16 bits per sample. Finally, the mean of each of the recorded sentences was removed from the recordings, so as to eliminate the effects of any DC biasing in the recording equipment.

3.3   Experimental details

To test the pitch estimation accuracy of the selected pitch estimation methods, first reference pitch estimates were generated from the individual utterances in the database, by use of the super-resolution pitch estimator detailed in [22]. This reference pitch estimator also generated voiced/unvoiced decisions. Fig 3.1 shows the reference pitch estimates generated for sentences A_2 and C_3 (sentence A spoken by speaker 2 and sentence C spoken by speaker 3) as an example. Then a database of four, two-speaker utterances was constructed by individually summing the utterances of sentences A, D, E, and F spoken by speaker 2 with the utterance of sentence C spoken by speaker 3. This produced four composite sentences: $A_2 + C_3, D_2 + C_3, E_2 + C_3$, and $F_2 + C_3$, where the subscript indicates the speaker number.

3.3.1   Experiment 1: pitch estimation at varying VVRs

In this experiment, the goal was to evaluate the performance of the different pitch estimators under varying voice to voice ratios (VVRs). To this end, first the samples
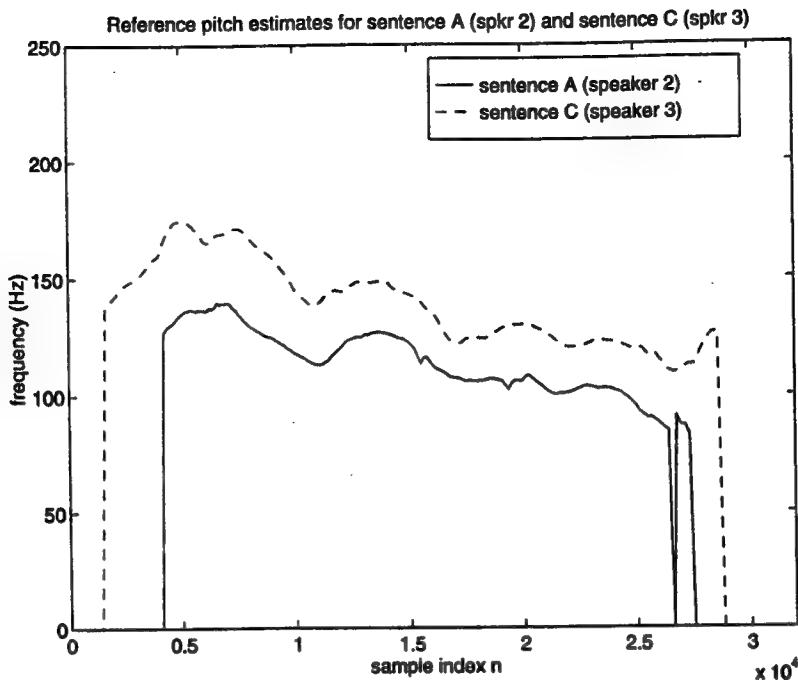
Figure 3.1: Reference pitch estimates generated for sentence A spoken by speaker 2 (A_2) and sentence C spoken by speaker 3 (C_3).

of utterance C_3 were scaled so that the variance over the entire utterance was equal to unity. Then the samples of utterances A_2, D_2, E_2, and F_2 were scaled such that the resulting composite sentences $A_2 + C_3, D_2 + C_3, E_2 + C_3$, and $F_2 + C_3$, produced by summation of the appropriate utterances, would have a VVR of 0 dB. These four composite sentences were then fed into each of the pitch estimation algorithms, which generated two pitch estimates for each input frame. The resulting pitch estimates were compared on a frame-by-frame basis with the reference pitches. Only those frames, in which both speakers' voices were marked as voiced by the reference pitch estimator, were used in the comparisons; pitch estimates generated for all other input frames were not counted. This procedure was then repeated for VVRs of 5 dB, 10 dB, and 15 dB.

3.3.2 Experiment 2: pitch estimation under degraded conditions

In this experiment, the goal was to evaluate the performance of the different pitch estimators when the co-channel speech signal has undergone degradations due to channel

distortions or additive noise. Such types of distortions are typically encountered in various communications scenarios. First the composite sentences $A_2 + C_3, D_2 + C_3, E_2 + C_3$, and $F_2 + C_3$ were produced by summation of the constituent sentences. The samples of each individual sentence were scaled to equal (unit) variance, so that resulting composite sentences had a 0 dB VVR.

To simulate the effects of channel distortions, the composite sentences were filtered with an FIR filter channel simulator which models the frequency response of a typical Continental Mid-quality Voice (CMV) phone line [15], featuring moderate low-frequency and high-frequency attenuation. The resulting degraded composite sentences were then fed into each of the pitch estimation algorithms, which generated two pitch estimates for each input frame. Again, the resulting pitch estimates were compared on a frame-by-frame basis with the reference pitches. Only those frames in which both speakers' voices were marked as voiced by the reference pitch estimator were used in the comparisons; pitch estimates generated for all other input frames were not counted. The above procedure was then repeated with a Continental Poor-quality Voice (CPV) channel simulator filter [15], which features more drastic attenuation of low and high frequencies than the CMV channel. The (magnitude) frequency response of the two channel simulators is shown in Fig. 3.2.

To investigate the effects of additive noise, white noise was added to each of the clean composite sentences so that the resulting SNRs were 10 dB. As with the channel-degraded speech, the resulting noise-degraded composite sentences were then fed into each of the pitch estimation algorithms, which generated two pitch estimates for each input frame. Comparison of the resulting pitch estimates with the reference pitch estimates proceeded in the same fashion as that described above for the channel-degraded speech.
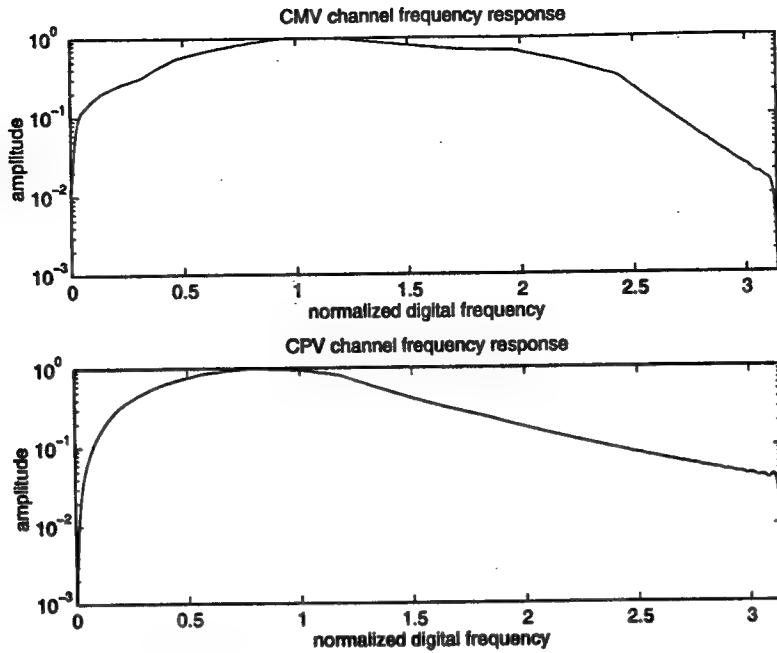
Figure 3.2: Frequency response of CMV (top) and CPV (bottom) channel simulator filters.

3.4 Experimental results

3.4.1 Experiment 1: pitch estimation at varying VVRs

Figs. 3.3–3.7 show the pitch estimates generated by the individual estimators on the composite sentence $\{A_1 + C_3\}$ at 0 dB VVR; the reference pitch tracks are shown in Fig. 3.1. Visual inspection and comparison of these graphs seems to indicate that the estimators are, for the most part, tracking the two speakers' pitches fairly closely. To quantify the results, first all pitch estimates are converted to an octave scale relative to 110 Hz to facilitate comparison. Then the first and second pitch estimates for each frame are individually compared with the closest reference pitch estimate for that frame. Figs. 3.9–3.14 show histograms of the pitch errors made by each of the estimators, accumulated over all testing utterances at 0 dB VVR. These results are summarized quantitatively in Table 3.2, which shows the percentage of frames in which the first and second pitch estimates fell greater than 10%, 3%, 2%, and 1% of an octave from the closest reference pitch estimate for that frame; i.e. the percentage of frames in which

A-55

the *errors* in the pitch estimates were greater than 10%, 3%, 2%, and 1% of an octave. In the case of 5 dB, 10 dB, and 15 dB VVRs, the first pitch estimate typically reflects the pitch of the louder speaker, while the second pitch estimate corresponds to the quieter speaker. The "total" column indicates the error percentages when the first and second pitch estimate errors are considered collectively. In that accurate determination of each speaker's pitch is a necessary, but not sufficient, condition for effective speaker separation, smaller percentage figures are better.

A typical error made by pitch estimators is that of *octave errors*, in which a multiple or sub-multiple of the actual pitch period is incorrectly identified as the pitch period. Such errors are reflected in the error histograms by the counts clustered around integer values of the abscissa. Table 3.3 shows the results of the same experiment after octave errors have been normalized; i.e. pitch estimates resulting in errors of greater than 1 octave are adjusted by adding or subtracting an integer number of octaves so as to bring the resulting pitch estimate within one octave of the reference pitch.
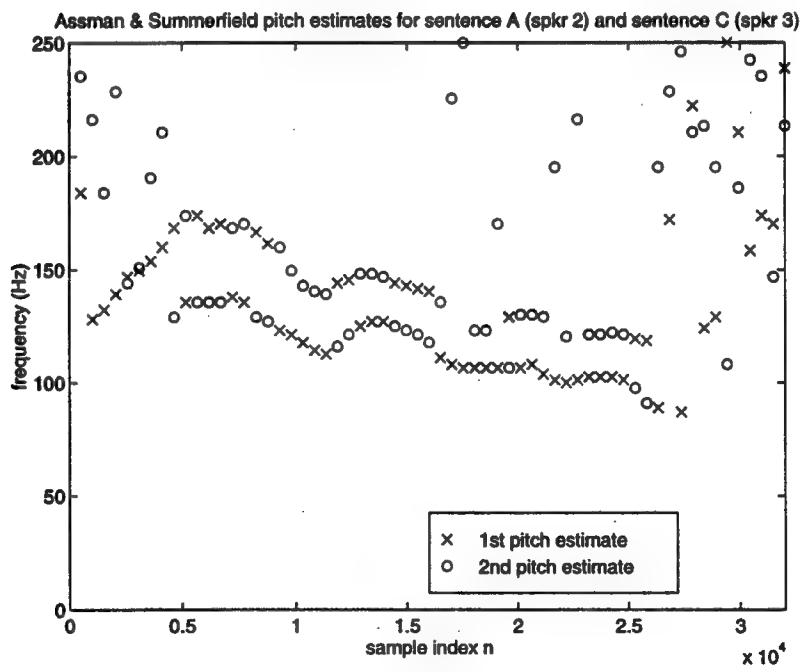
Figure 3.3: Pitch estimates generated by Assman and Summerfield's pitch estimator for composite sentence $A_1 + C_3$ (VVR=0 dB).
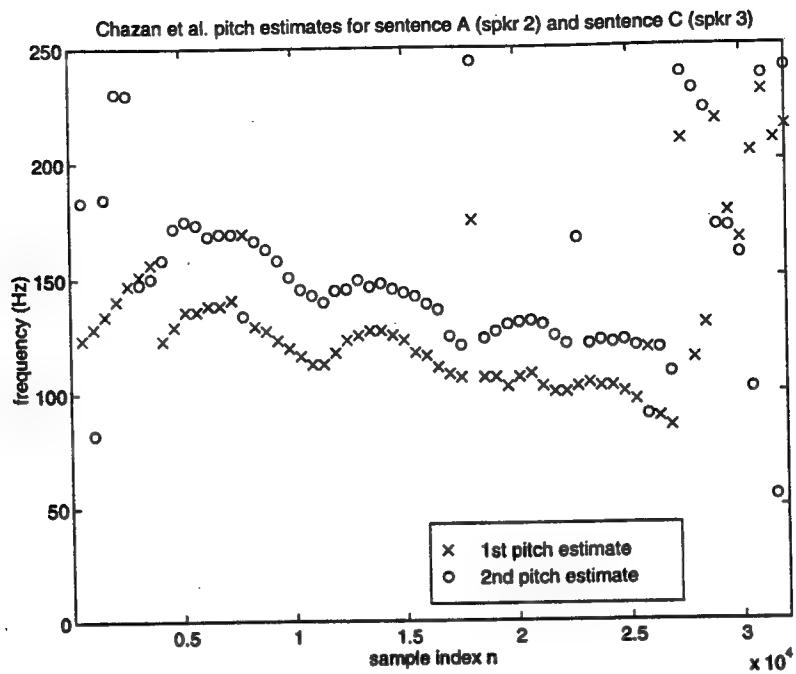
Figure 3.4: Pitch estimates generated by Chazan et al.'s pitch estimator for composite sentence $A_1 + C_3$ (VVR=0 dB).
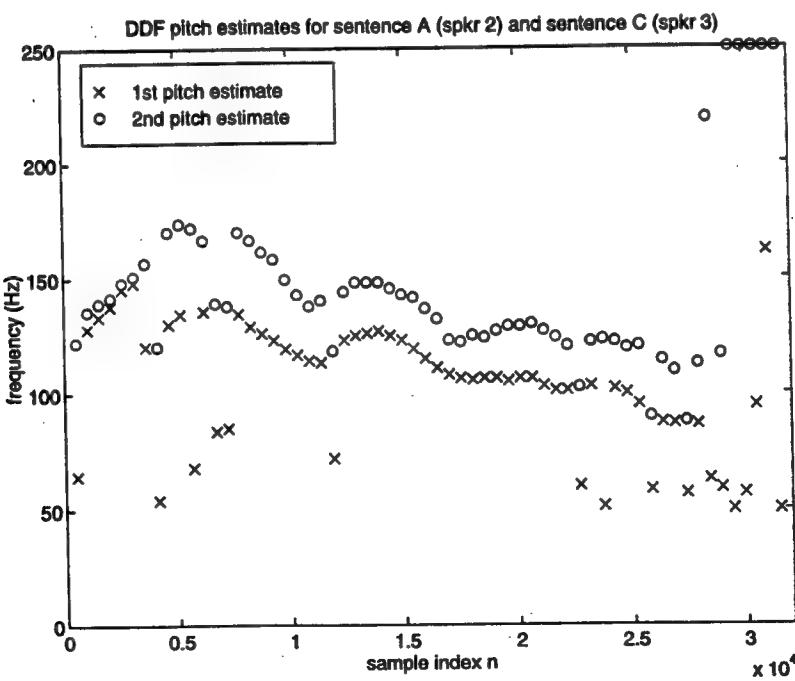
Figure 3.5: Pitch estimates generated by de Cheveigné's DDF pitch estimator for composite sentence $A_1 + C_3$ (VVR=0 dB).
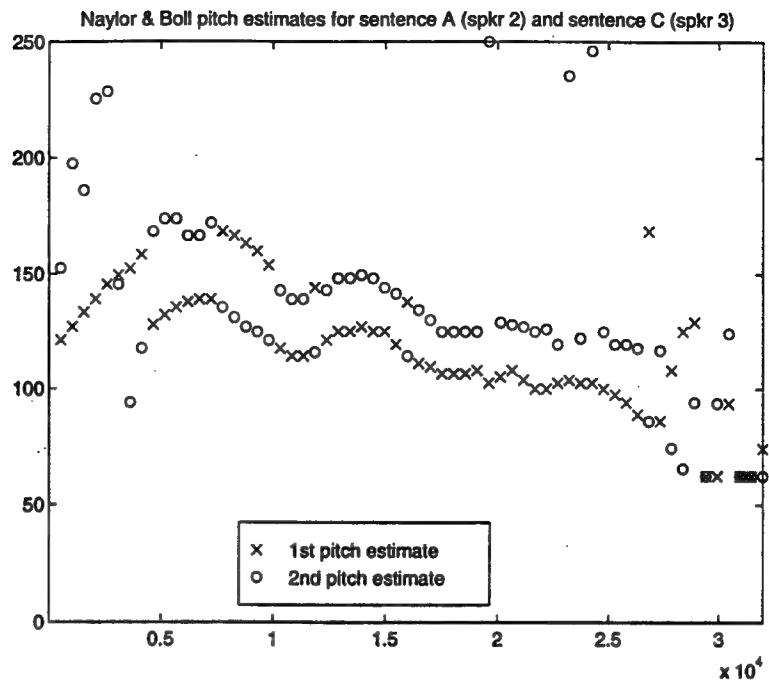
Figure 3.6: Pitch estimates generated by Maximum Likelihood pitch estimator used by Naylor and Boll for composite sentence $A_1 + C_3$ (VVR=0 dB).
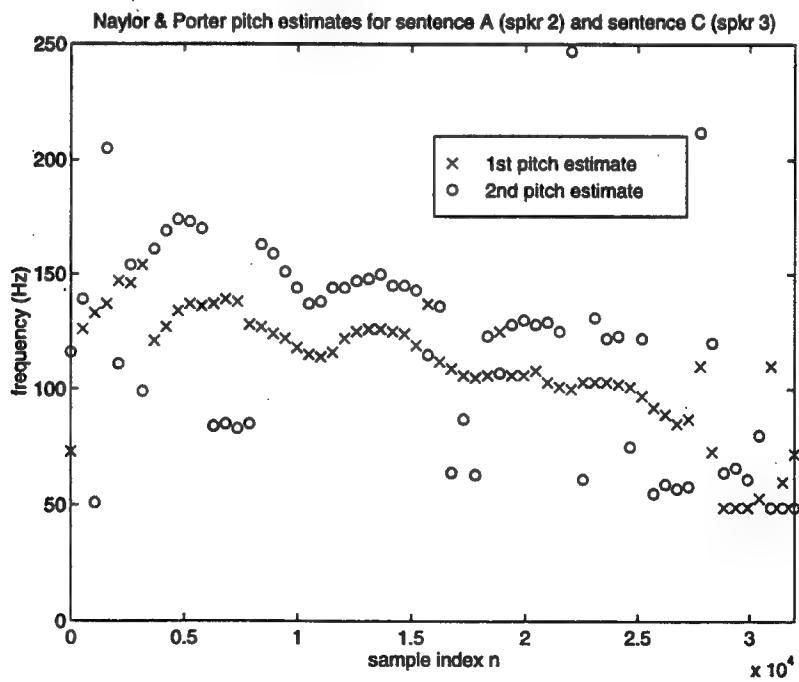

Figure 3.7: Pitch estimates generated by Naylor and Porter's pitch estimator for composite sentence $A_1 + C_3$ (VVR=0 dB).
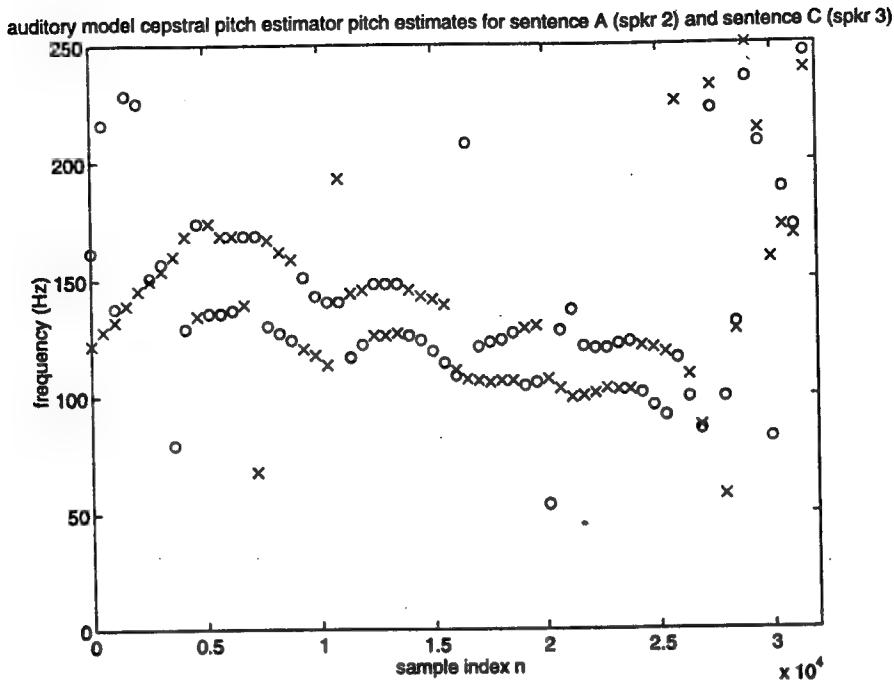
Figure 3.8: Pitch estimates generated by auditory model cepstral pitch estimator for composite sentence $A_1 + C_3$ (VVR=0 dB).

algorithm	VVR	> 10%			> 3%			> 2%			> 1%		
		1st	2nd	total	1st	2nd	total	1st	2nd	total	1st	2nd	total
Assman & Summerfield	0 dB	4%	18%	11%	7%	22%	15%	13%	31%	22%	39%	52%	45%
"	5 dB	4%	25%	14%	7%	32%	19%	15%	39%	27%	40%	59%	49%
"	10 dB	4%	39%	22%	7%	51%	29%	13%	61%	37%	36%	75%	56%
"	15 dB	5%	55%	30%	8%	69%	38%	15%	74%	45%	37%	86%	61%
Chazan et al.	0 dB	7%	16%	12%	16%	22%	19%	30%	35%	32%	56%	64%	60%
"	5 dB	6%	26%	16%	17%	36%	27%	31%	46%	39%	58%	67%	62%
"	10 dB	9%	45%	27%	15%	58%	36%	29%	64%	46%	54%	78%	66%
"	15 dB	9%	52%	30%	17%	66%	42%	30%	76%	53%	56%	89%	72%
de Cheveigné	0 dB	2%	22%	12%	9%	27%	18%	21%	36%	29%	51%	62%	57%
"	5 dB	2%	29%	16%	12%	35%	23%	27%	42%	34%	55%	63%	59%
"	10 dB	2%	38%	20%	15%	42%	29%	31%	49%	40%	58%	69%	63%
"	15 dB	9%	35%	22%	32%	45%	39%	50%	50%	49%	68%	72%	70%
Naylor & Boll	0 dB	4%	10%	7%	9%	21%	15%	13%	30%	22%	41%	55%	48%
"	5 dB	3%	23%	13%	5%	35%	20%	9%	47%	28%	34%	64%	49%
"	10 dB	2%	37%	20%	5%	59%	32%	7%	74%	41%	30%	82%	56%
"	15 dB	2%	54%	28%	4%	80%	42%	6%	87%	46%	29%	92%	60%
Naylor & Porter	0 dB	6%	33%	19%	13%	41%	27%	24%	51%	37%	50%	68%	59%
"	5 dB	5%	57%	31%	16%	67%	42%	25%	75%	50%	52%	88%	70%
"	10 dB	4%	82%	43%	11%	90%	51%	21%	93%	57%	53%	96%	75%
"	15 dB	4%	92%	48%	12%	98%	55%	26%	99%	62%	56%	99%	78%
auditory-cepstral	0 dB	6%	10%	8%	8%	18%	13%	16%	26%	21%	34%	45%	39%
"	5 dB	4%	15%	9%	7%	27%	17%	13%	36%	25%	28%	55%	41%
"	10 dB	2%	18%	10%	6%	38%	22%	9%	50%	29%	26%	65%	46%
"	15 dB	4%	28%	16%	6%	58%	32%	10%	70%	40%	27%	81%	54%

Table 3.2: Percentage of estimates that fell further than 10%, 3%, or 1% of an octave (relative to 110 Hz) from the reference pitch estimate.

A-59

algorithm	VVR	> 10%			> 3%			> 2%			> 1%		
		1st	2nd	total	1st	2nd	total	1st	2nd	total	1st	2nd	total
Assman & Summerfield	0 dB	3%	13%	8%	6%	22%	14%	13%	31%	22%	39%	52%	45%
"	5 dB	2%	18%	10%	6%	30%	18%	15%	38%	26%	40%	58%	49%
"	10 dB	2%	29%	16%	6%	49%	28%	13%	59%	36%	36%	75%	56%
"	15 dB	4%	46%	25%	8%	66%	37%	15%	72%	43%	37%	84%	61%
Chazan et al.	0 dB	4%	10%	7%	15%	21%	18%	29%	35%	32%	55%	64%	59%
"	5 dB	2%	19%	11%	15%	35%	25%	29%	46%	37%	56%	67%	61%
"	10 dB	5%	35%	20%	12%	53%	32%	26%	62%	44%	51%	76%	63%
"	15 dB	6%	44%	25%	16%	64%	40%	29%	73%	51%	55%	87%	71%
de Cheveigné	0 dB	1%	16%	9%	9%	22%	16%	21%	30%	26%	51%	58%	54%
"	5 dB	2%	27%	14%	12%	33%	22%	27%	41%	34%	55%	62%	58%
"	10 dB	2%	36%	19%	15%	41%	28%	31%	49%	40%	58%	68%	63%
"	15 dB	7%	32%	20%	32%	44%	38%	47%	49%	48%	67%	72%	70%
Naylor & Boll	0 dB	3%	6%	5%	7%	18%	13%	12%	29%	20%	41%	53%	47%
"	5 dB	2%	16%	9%	4%	31%	18%	8%	44%	26%	33%	61%	47%
"	10 dB	2%	31%	17%	5%	56%	31%	7%	72%	39%	30%	79%	55%
"	15 dB	2%	47%	25%	4%	78%	41%	6%	86%	46%	29%	91%	60%
Naylor & Porter	0 dB	5%	31%	18%	12%	40%	26%	23%	50%	36%	50%	67%	59%
"	5 dB	4%	53%	29%	16%	67%	41%	24%	75%	49%	52%	88%	70%
"	10 dB	4%	79%	41%	11%	89%	50%	21%	92%	57%	53%	95%	74%
"	15 dB	4%	87%	46%	12%	96%	54%	26%	98%	62%	56%	98%	77%
auditory-cepstral	0 dB	2%	7%	5%	6%	15%	10%	15%	24%	19%	33%	44%	39%
"	5 dB	2%	12%	7%	6%	26%	16%	12%	35%	24%	28%	55%	41%
"	10 dB	2%	13%	8%	5%	36%	21%	9%	49%	29%	26%	64%	45%
"	15 dB	4%	21%	12%	6%	56%	31%	10%	68%	39%	27%	79%	53%

Table 3.3: Percentage of estimates that fell further than 10%, 3%, or 1% of an octave (relative to 110 Hz) from the reference pitch estimate after octave errors have been normalized.

3.4.2 Experiment 2: pitch estimation under degraded conditions

Table 3.4 shows the percentage of frames, over all testing utterances, in which the pitch errors were greater than 10%, 3%, 2%, and 1% of an octave when the 0 dB VVR composite sentences were degraded by the CMV channel simulator. Table 3.5 shows the same results, after octave errors have been normalized.

Table 3.6 shows the percentage of frames, over all testing utterances, in which the pitch errors were greater than 10%, 3%, 2%, and 1% of an octave when the 0 dB VVR composite sentences were degraded by the more severe CPV channel simulator. Table 3.7 shows the same results, after octave errors have been normalized.

Table 3.8 shows the percentage of frames, over all testing utterances, in which the pitch errors were greater than 10%, 3%, 2%, and 1% of an octave when the 0 dB VVR composite sentences were degraded by white noise at a SNR of 10 dB. Table 3.9 shows the same results, after octave errors have been normalized.

algorithm	VVR	> 10%			> 3%			> 2%			> 1%		
		1st	2nd	total	1st	2nd	total	1st	2nd	total	1st	2nd	total
Assman & Summerfield	0 dB	7%	19%	13%	9%	27%	18%	15%	33%	24%	36%	58%	47%
Chazan et al.	0 dB	25%	32%	28%	29%	40%	34%	33%	44%	39%	56%	68%	62%
de Cheveigné	0 dB	1%	18%	10%	4%	19%	11%	6%	13%	21%	17%	29%	23%
Naylor & Boll	0 dB	10%	12%	11%	12%	24%	18%	18%	35%	26%	36%	54%	45%
Naylor & Porter	0 dB	5%	24%	14%	19%	43%	31%	32%	52%	42%	59%	74%	67%
auditory-cepstral	0 dB	5%	11%	8%	7%	23%	15%	13%	30%	21%	30%	50%	40%

Table 3.4: Percentage of estimates that fell further than 10%, 3%, or 1% of an octave (relative to 110 Hz) from the reference pitch estimate when the speech has passed through the CMV channel.

algorithm	VVR	> 10%			> 3%			> 2%			> 1%		
		1st	2nd	total	1st	2nd	total	1st	2nd	total	1st	2nd	total
Assman & Summerfield	0 dB	5%	15%	10%	9%	26%	18%	15%	33%	24%	36%	58%	47%
Chazan et al.	0 dB	18%	25%	22%	25%	38%	32%	30%	44%	37%	53%	68%	61%
de Cheveigné	0 dB	1%	13%	7%	4%	14%	9%	6%	16%	11%	17%	24%	21%
Naylor & Boll	0 dB	7%	9%	8%	10%	23%	16%	16%	35%	25%	35%	53%	44%
Naylor & Porter	0 dB	5%	21%	13%	19%	42%	31%	32%	51%	42%	59%	73%	66%
auditory-cepstral	0 dB	4%	10%	7%	7%	22%	15%	13%	30%	21%	30%	50%	40%

Table 3.5: Percentage of estimates that fell further than 10%, 3%, or 1% of an octave (relative to 110 Hz) from the reference pitch estimate when the speech has passed through the CMV channel and after octave errors have been normalized.

algorithm	VVR	> 10%			> 3%			> 2%			> 1%		
		1st	2nd	total	1st	2nd	total	1st	2nd	total	1st	2nd	total
Assman & Summerfield	0 dB	6%	22%	14%	9%	30%	19%	15%	35%	25%	35%	57%	46%
Chazan et al.	0 dB	27%	39%	33%	33%	45%	39%	38%	51%	44%	57%	68%	62%
de Cheveigné	0 dB	4%	20%	12%	7%	21%	14%	9%	22%	16%	19%	31%	25%
Naylor & Boll	0 dB	15%	16%	16%	19%	31%	25%	21%	37%	29%	38%	61%	49%
Naylor & Porter	0 dB	2%	29%	16%	17%	47%	32%	28%	55%	41%	61%	72%	67%
auditory-cepstral	0 dB	6%	16%	11%	8%	29%	19%	14%	36%	25%	29%	55%	42%

Table 3.6: Percentage of estimates that fell further than 10%, 3%, or 1% of an octave (relative to 110 Hz) from the reference pitch estimate when the speech has passed through the CPV channel.

algorithm	VVR	> 10%			> 3%			> 2%			> 1%		
		1st	2nd	total	1st	2nd	total	1st	2nd	total	1st	2nd	total
Assman & Summerfield	0 dB	4%	15%	10%	9%	28%	18%	15%	34%	24%	35%	56%	46%
Chazan et al.	0 dB	19%	33%	26%	29%	41%	35%	34%	48%	41%	55%	67%	61%
de Cheveigné	0 dB	4%	15%	9%	7%	16%	12%	9%	17%	13%	19%	26%	22%
Naylor & Boll	0 dB	9%	13%	11%	13%	29%	21%	17%	36%	26%	35%	60%	48%
Naylor & Porter	0 dB	2%	24%	13%	17%	45%	31%	28%	53%	41%	61%	71%	66%
auditory-cepstral	0 dB	6%	14%	10%	8%	28%	18%	14%	36%	25%	29%	55%	42%

Table 3.7: Percentage of estimates that fell further than 10%, 3%, or 1% of an octave (relative to 110 Hz) from the reference pitch estimate when the speech has passed through the CPV channel and after octave errors have been normalized.
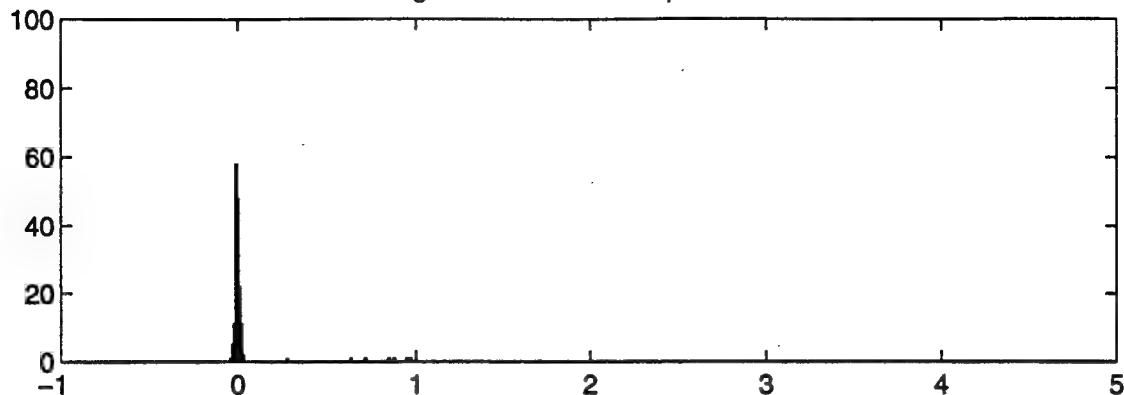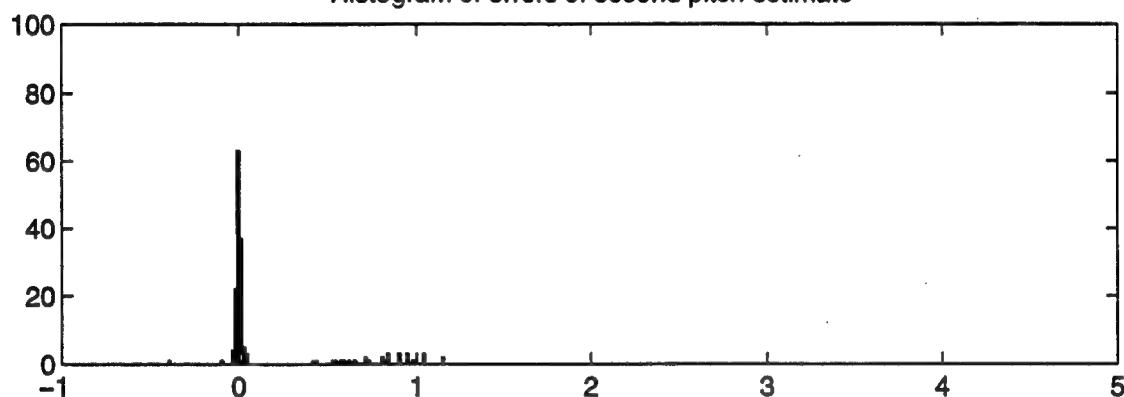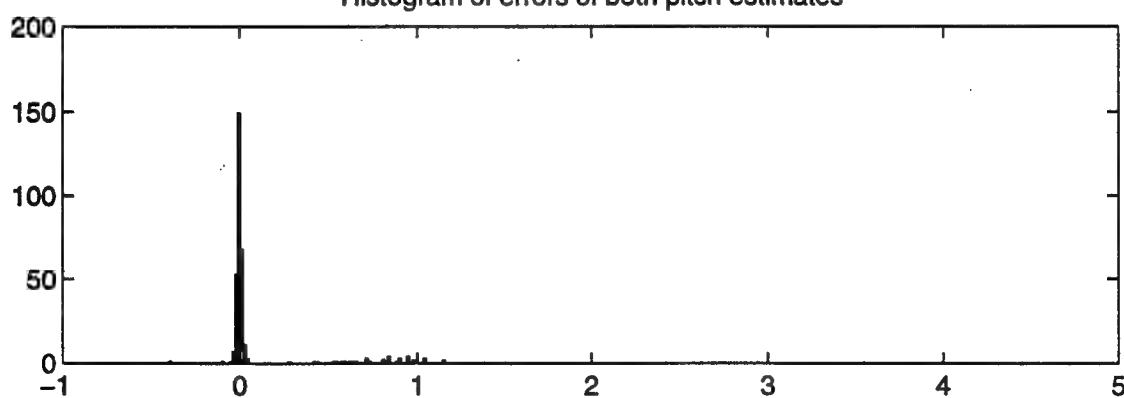
Figure 3.9: Histogram of pitch errors of Assman and Summerfield's pitch estimator (VVR=0 dB).

A-62
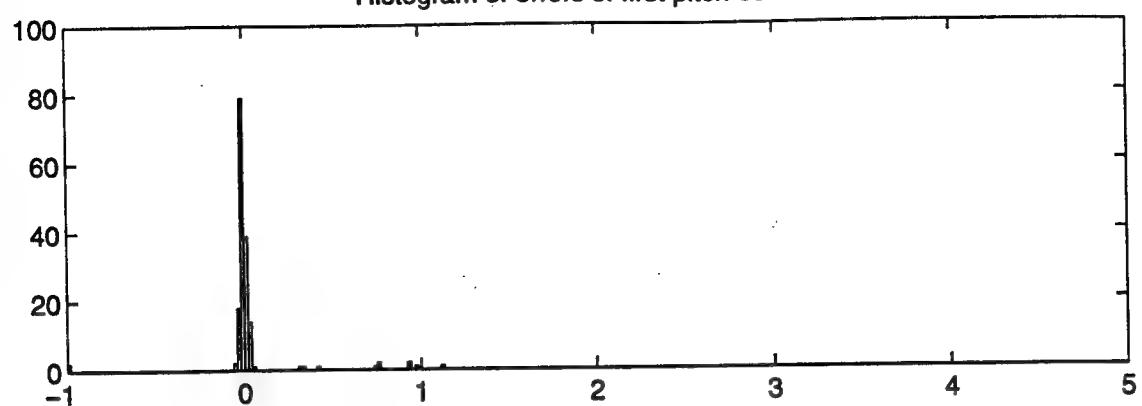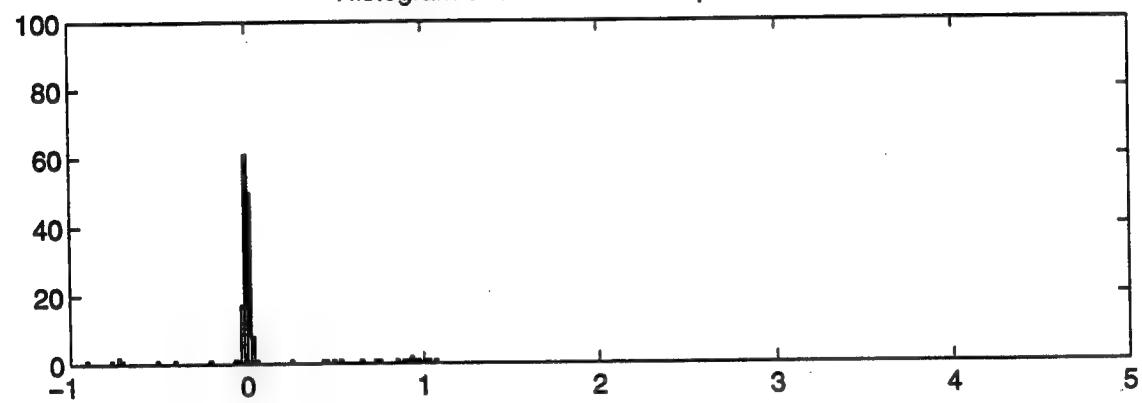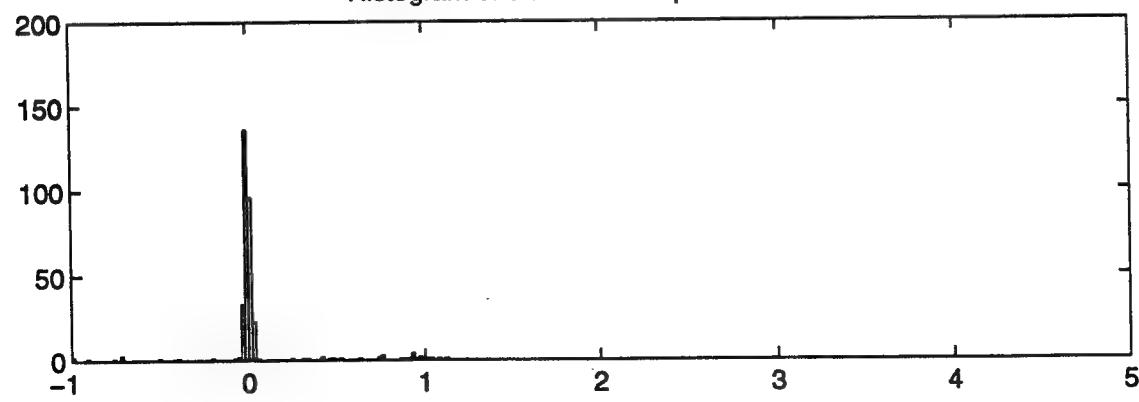
Figure 3.10: Histogram of pitch errors of Chazan et al.'s pitch estimator (VVR=0 dB).
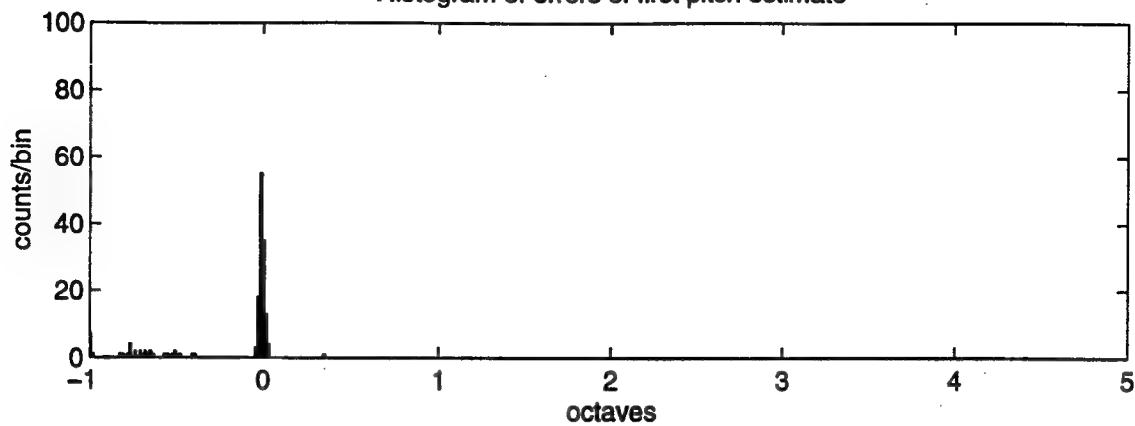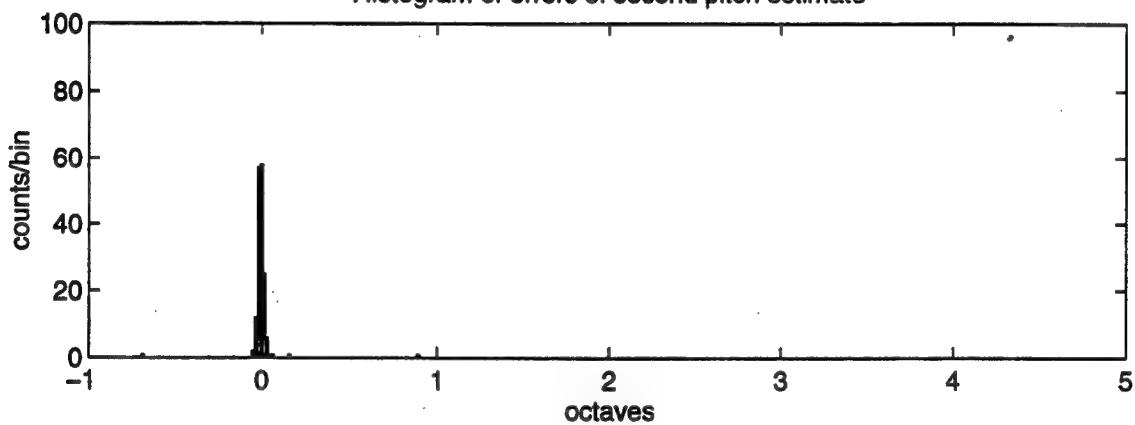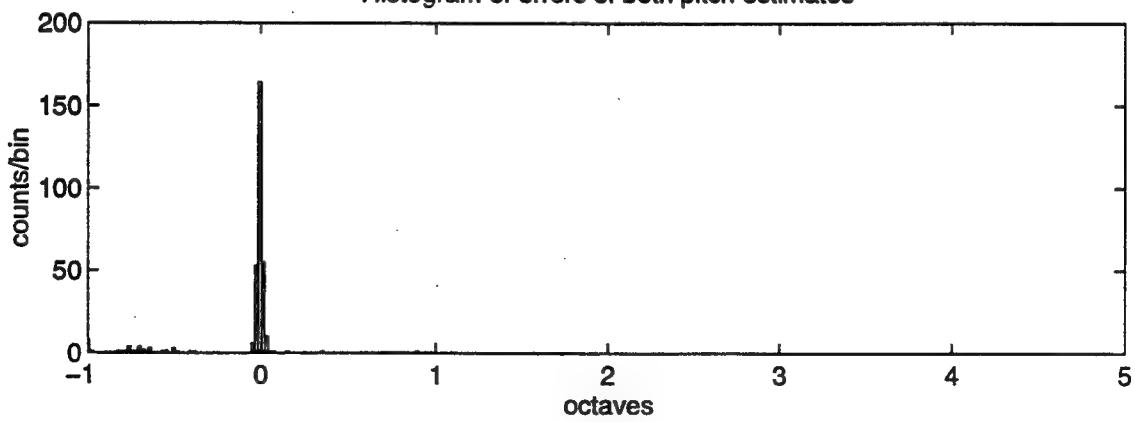
Figure 3.11: Histogram of pitch errors of de Cheveigné's DDF pitch estimator (VVR=0 dB).
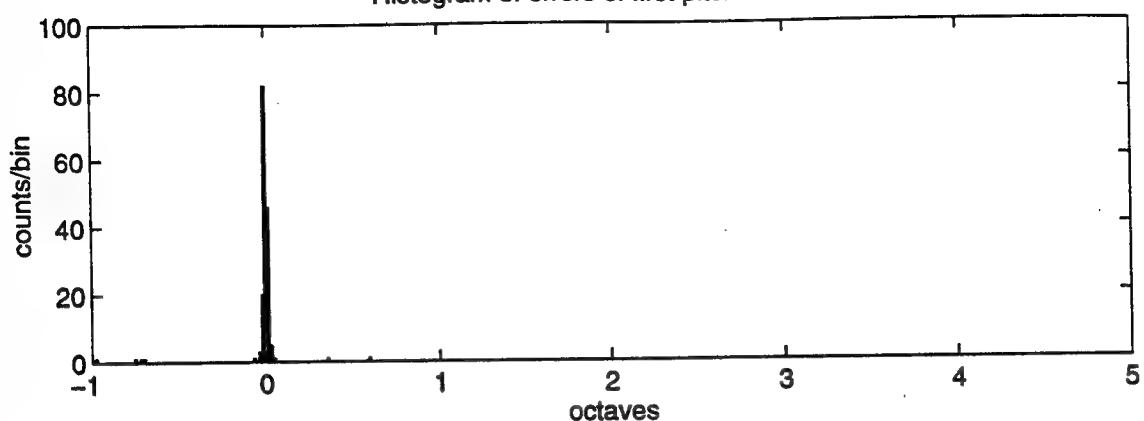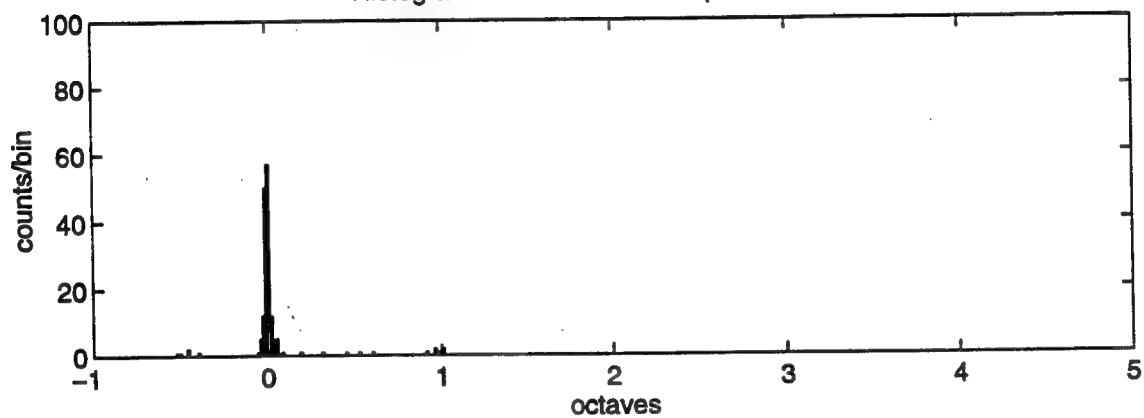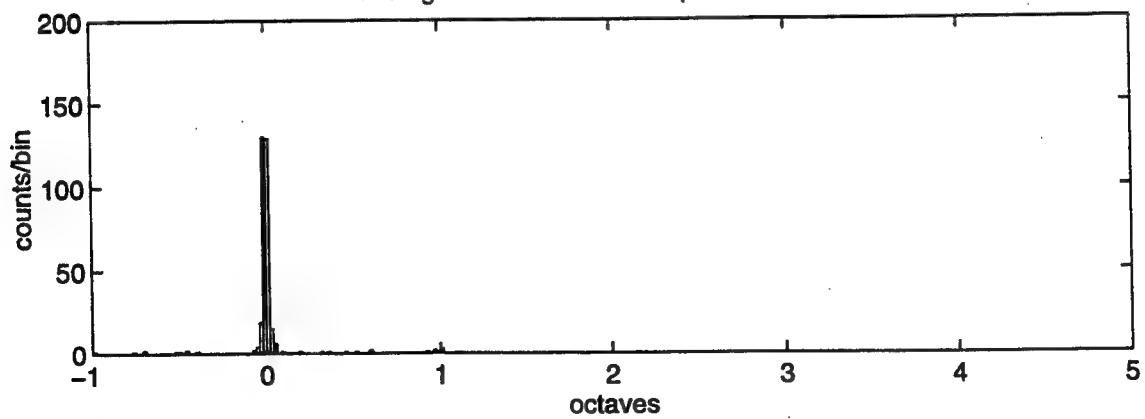
Figure 3.12: Histogram of pitch errors of Maximum Likelihood pitch estimator used by Naylor and Boll (VVR=0 dB).
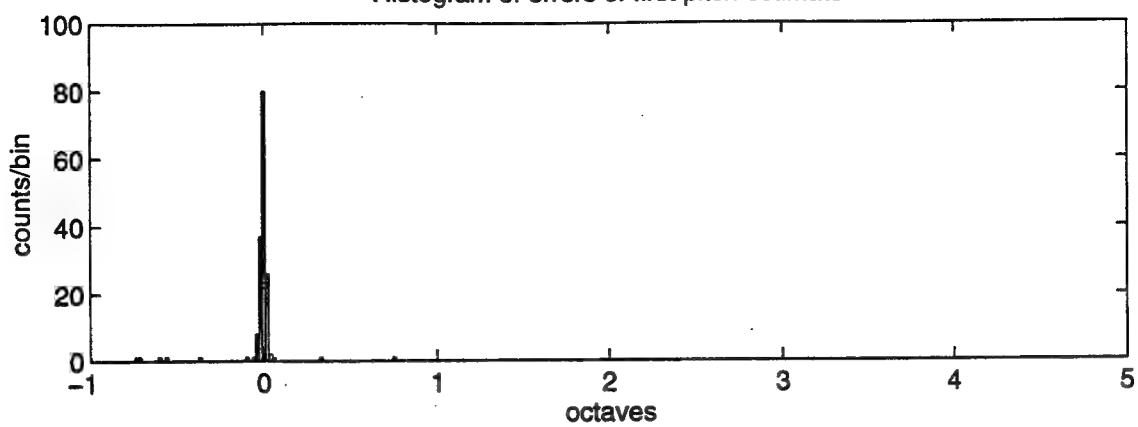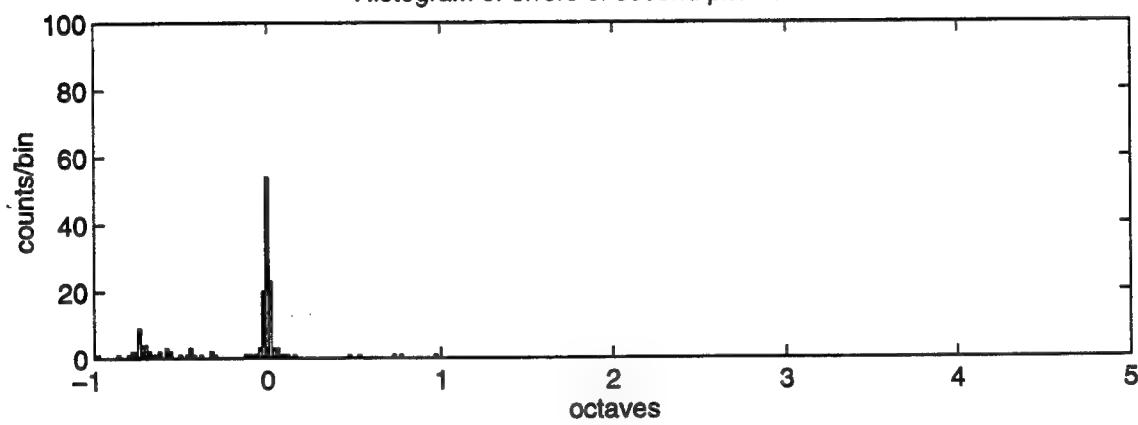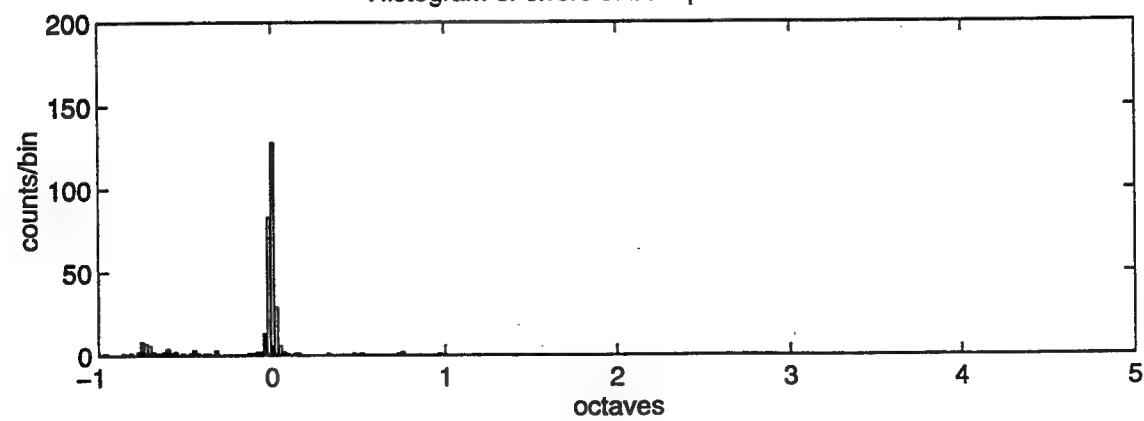
Figure 3.13: Histogram of pitch errors of Naylor and Porter's pitch estimator (VVR=0 dB).
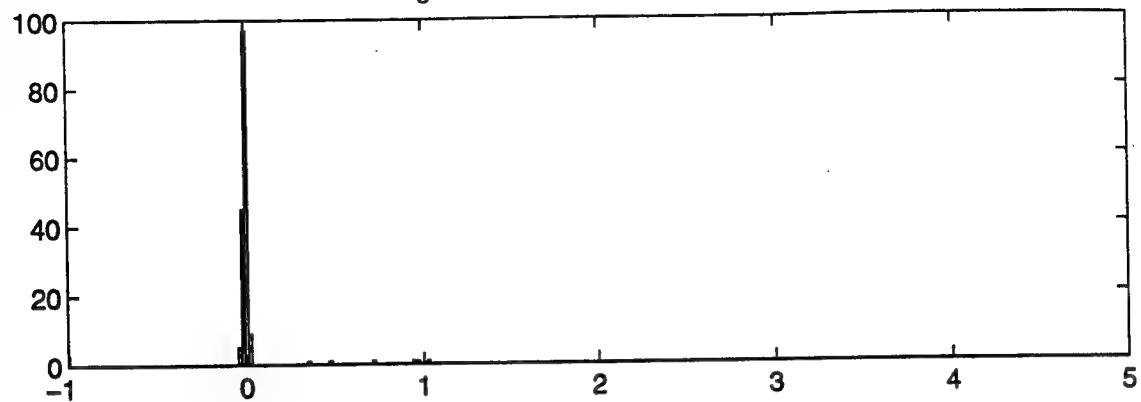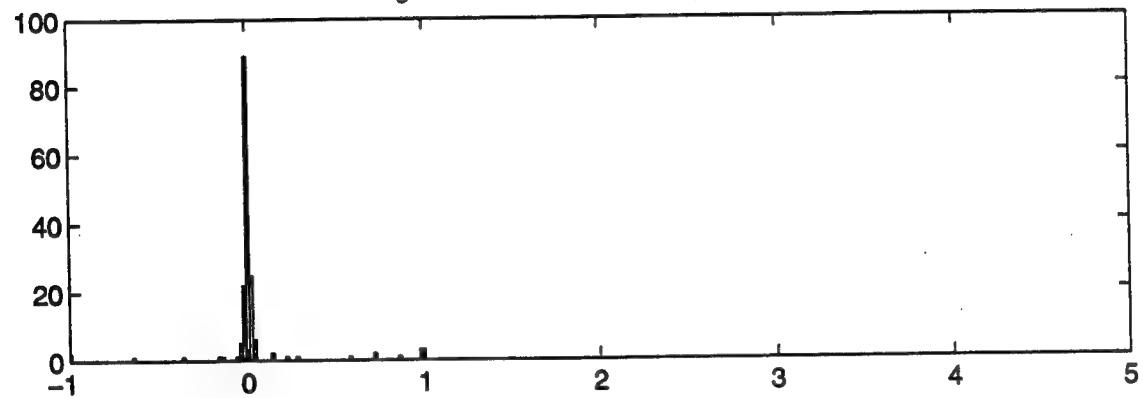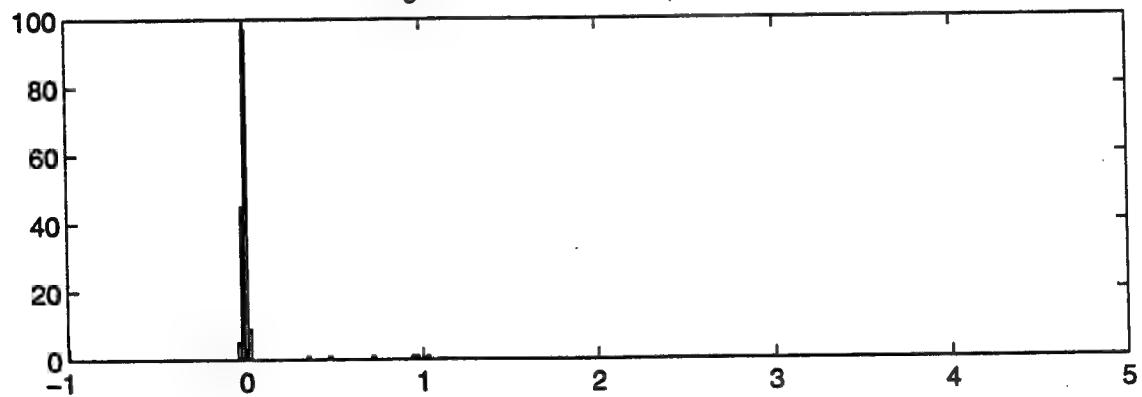
Figure 3.14: Histogram of pitch errors of the auditory model cepstral pitch estimator (VVR=0 dB).

algorithm	VVR	> 10%			> 3%			> 2%			> 1%		
		1st	2nd	total	1st	2nd	total	1st	2nd	total	1st	2nd	total
Assman & Summerfield	0 dB	29%	41%	35%	39%	52%	45%	45%	64%	55%	66%	77%	72%
Chazan et al.	0 dB	64%	71%	67%	77%	79%	78%	81%	85%	83%	87%	93%	90%
de Cheveigné	0 dB	2%	29%	16%	12%	36%	24%	24%	44%	34%	53%	65%	59%
Naylor & Boll	0 dB	5%	9%	7%	8%	19%	13%	13%	29%	21%	44%	54%	49%
Naylor & Porter	0 dB	6%	35%	21%	17%	47%	32%	29%	56%	42%	55%	73%	64%
auditory-cepstral	0 dB	30%	38%	34%	36%	55%	45%	44%	63%	54%	57%	79%	68%

Table 3.8: Percentage of estimates that fell further than 10%, 3%, or 1% of an octave (relative to 110 Hz) from the reference pitch estimate when the speech has been corrupted by 10 dB white noise.

algorithm	VVR	> 10%			> 3%			> 2%			> 1%		
		1st	2nd	total	1st	2nd	total	1st	2nd	total	1st	2nd	total
Assman & Summerfield	0 dB	25%	32%	28%	38%	48%	43%	44%	61%	52%	66%	76%	71%
Chazan et al.	0 dB	48%	54%	51%	69%	73%	71%	75%	81%	78%	84%	90%	87%
de Cheveigné	0 dB	2%	21%	12%	12%	29%	20%	24%	37%	30%	53%	58%	56%
Naylor & Boll	0 dB	4%	6%	5%	7%	16%	12%	12%	28%	20%	44%	53%	48%
Naylor & Porter	0 dB	6%	32%	19%	17%	44%	30%	29%	53%	41%	55%	71%	63%
auditory-cepstral	0 dB	27%	33%	30%	33%	52%	42%	42%	62%	52%	55%	78%	66%

Table 3.9: Percentage of estimates that fell further than 10%, 3%, or 1% of an octave (relative to 110 Hz) from the reference pitch estimate when the speech has been corrupted by 10 dB white noise and after octave errors have been normalized.

Chapter 4

Conclusion

4.1 Discussions

4.1.1 Experiment 1: pitch estimation at varying VVRs

- The auditory model cepstral pitch estimator generated fewer total pitch errors than all other methods, across all VVRs tested, and for all error thresholds, 1% 2%, 3%, and 10% of an octave; the one exception was the percentage of total errors > 10% at 0 dB VVR, where it scored 1% more errors than the lowest scoring estimator for that case.

- As the VVR was increased, the error rates for some of the pitch estimators' first pitch estimates *increased*; this is surprising in that as the VVR increases, the interference due to the secondary speaker should decrease, making the primary speaker's pitch more readily detectable.

- For increasing VVRs, the error rates of the secondary pitch estimates increased as well. This was to be expected, since a positive VVR means that the power of the secondary speech signal is less than that of the primary one.

- The auditory model cepstral pitch estimator performed significantly better than the estimator of Assman and Summerfield, which is also based upon an auditory model, especially at the 1% octave error threshold and VVRs of 5 dB, 10 dB, and 15 dB.

- All pitch estimation methods were shown to be prone to some octave errors; however, the similarity of Tables 3.2 and 3.3 indicates that none of the pitch estimators experienced an appreciable amount of such errors.

A-69

4.1.2 Experiment 2: pitch estimation under degraded conditions

- In the case of degradation by the CMV channel, the DDF method of de Cheveigné outperformed all other methods, by roughly a factor of two.

- For the CMV channel degradation, the auditory model cepstral pitch estimator generated the second fewest errors. The pitch estimator of Assman and Summerfield and the Maximum Likelihood pitch estimator used by Naylor and Boll performed about the same, generating the third fewest errors, followed by Chazan et al.'s and Naylor and Porter's pitch estimators, both operating at roughly three times the error rate of the best performing DDF estimator.

- Performance of the pitch estimators was roughly equal in the case of the CMV channel and in the case of no channel (and 0 dB VVR). The one striking exception was de Cheveigné's DDF estimator, whose error rates *decreased* by roughly a factor of two. It may be that the smoothing of the time domain waveform, afforded by the high-frequency attenuation of the CMV channel, resulted in less sensitivity of the ACF-based DDF pitch estimator. It is common practice to smooth a signal with low-pass filter prior to pitch analysis via the ACF, so as to reduce the effects of high frequencies which tend to reduce correlation values and spuriously bias the resulting pitch estimates. A second, less striking exception was the pitch estimator of Naylor and Porter, whose error rates increased in the CMV channel case by 8% in total error rate at the 1% octave threshold.

- Degradation by the CMV channel did not seem to have any significant effect on the percentage of octave errors made by any of the pitch estimators; this is reflected in the similarity of Tables 3.4 and 3.5. This was somewhat surprising, in that it was thought that the roll-off of the CMV filter at the low frequency end might cause attenuation of the fundamental frequency in some speech signals, which could possibly result in more octave errors being made.

- When the co-channel signal has been degraded by the more severe CPV channel, de Cheveigné's DDF method emerged as the best performer, again scoring error

rates roughly half that of the next best performing algorithm.

- For the CPV channel, the auditory model cepstral pitch estimator again generated the second fewest errors, followed by the estimators of Assman and Summerfield and the Maximum Likelihood pitch estimator used by Naylor and Boll, and finally Chazan et al.'s and Naylor and Porter's estimators.

- Performance of the pitch estimators was roughly equal in the case of the CPV channel and in the case of no channel (and 0 dB VVR). The notable exception again was de Cheveigné's pitch estimator, whose error rates again dropped by a factor of two with respect to the no channel case. Again, this improvement can probably be attributed to the time-domain smoothing afforded by the high-frequency attenuation of the CPV filter. A second, less drastic exception was the pitch estimator of Naylor and Porter's, whose error rates increased in the CPV channel case by 7% in total error rate at the 1% octave threshold.

- Degradation by the CPV channel also did not seem to have any significant effect on the percentage of octave errors made by any of the pitch estimators; this is reflected in the similarity of Tables 3.6 and 3.7. Again, this was surprising, as the increased low-frequency attenuation of the CPV channel relative to the CMV channel would be expected to attenuate fundamental frequency components of input signals even more so.

- In the case of additive 10 dB white noise, as expected, all pitch estimators experienced performance degradation. The Maximum Likelihood pitch estimator employed by Naylor and Boll emerged as the best performer, scoring 10% fewer overall errors than the next best performing algorithm. The DDF pitch estimator and the Maximum Likelihood pitch estimator experienced almost no degradation with respect to the no noise case. Assman and Summerfield's pitch estimator, Chazan et al.'s pitch estimator and and the auditory model cepstral pitch estimator suffered significant performance degradations.

4.1.3 General discussions

- There were a total of 165 frames which were used in the computation of the error rates. Therefore, all figures have a granularity of $\frac{1}{165} = 0.61\%$.

- As pointed out several times in the review of the previous work, current systems for co-channel speaker separation suffer from a number of limitations. These include: 1) inability to process cases other than voiced speech on voiced speech, 2) inability to accurately detect the voicing of each speaker and the number of speakers speaking at a given instant, 3) inability to maintain continuity of speaker identity from frame to frame, 4) limited separation effectiveness due to pitch estimation inaccuracy and limited pitch estimate resolution, 5) limited separation effectiveness due to inaccurate modeling of the quasi-periodic nature of real voiced speech. While each of the proposed methods does not suffer from all of these limitations, all of the methods are subject to at least some of these limitations, and possibly others not mentioned here but discussed in Chapter 2. The main limitation faced by most previous work, however, is that usually, only the tasks of pitch estimation and signal separation were addressed; very few attempts were made to integrate these processing stages into complete systems.

4.2 Summary

- A comprehensive review of all major work on co-channel speaker separation was presented. It was shown that most speaker separation methods consisted of two main processing stages: 1) estimation of the speakers' pitches, and 2) separation of the speech signals using these pitch estimates.

- A new method for performing estimation of both speakers' pitches from the co-channel speech signal was developed.

- The performance of this pitch detection algorithm was tested under a variety of VVRs and signal degradations, including distortion due to frequency selective channels and additive noise. Performance was compared with the pitch estimation

algorithms employed in a number of speaker-separation studies.

- For clean co-channel speech, the auditory model cepstral pitch estimator emerged as the best overall performer, producing the lowest error rates at all tested VVRs, and across all error thresholds.

- For channel-degraded co-channel speech, de Cheveigné's DDF pitch estimator markedly outperformed all other estimators, typically by a factor of one half fewer errors.

- For noise-corrupted co-channel speech, the Maximum Likelihood pitch estimator employed by Naylor and Boll outperformed all others.

4.3   Future Work

- Most all previous work has focused on the voiced/voiced case, and has revolved around the estimation of the speakers' pitches from the co-channel signal. It appears that methodologies based on this type of pitch-based approach have been exhausted, and that only limited, if any, further progress can be made in this direction. New separation methods, based upon the formant structure of co-channel speech, rather than the pitch structure, need to be developed in order to be able to achieve separation of arbitrary types of speech combinations, not just voiced on voiced speech. Preliminary work in this direction is already underway.

- A critical, though not widely-addressed issue in co-channel speaker separation is the accurate determination of the number of speakers present and the voicing of those speakers. While a few studies have briefly addressed this problem, the brunt of the research has been on pitch estimation from the co-channel signal. However this processing step is necessary for ensuring that subsequent stages process the co-channel signal in a manner consistent with the nature of the voice signal(s) present.

References

[1] P. F. Assman and Q. Summerfield. Modeling the perception of concurrent vowels: Vowels with different fundamental frequencies. *Journal of the Acoustical Society of America*, 88(2):680–697, August 1990.

[2] S.F. Boll and R.E. Wohlford. Event driven speech enhancement. In *Proceedings ICASSP-1983*, pages 1152–1155, 1983.

[3] C. Chafe, D. Jaffe, K. Kashima, B. Mont-Reynaud, and J. Smith. Techniques for note identification in polyphonic music. In *Proceedings of the International Computer Music Conference (ICMC)-1985*, pages 399–405, 1985.

[4] Chris Chafe and David Jaffe. Source separation and note identification in polyphonic music. In *Proceedings ICASSP-1986*, pages 1289–1292, 1986.

[5] D. Chazan, Y. Stettiner, and D. Malah. Optimal multi-pitch estimation using the em algorithm for co-channel speech separation. In *Proceedings ICASSP-1993*, pages II–728–II–731, 1993.

[6] A. de Cheveigné. Separation of concurrent harmonic sounds: Fundamental frequency estimation and a time-domain cancellation model. *Journal of the Acoustical Society of America*, 93(6):3271–3290, June 1993.

[7] M. Feder and E. Weinstein. Parameter estimation of superimposed signals using the em algorithm. In *IEEE Transactions on Acoustics, Speech, and Signal Processing*, volume 36, pages 477–489, 1988.

[8] J. L. Flanagan, C. H. Coker, L. R. Rabiner, R. W. Schafer, and N. Umeda. Synthetic voices for computers. *IEEE Spectrum*, 7(10):22–45, October 1970.

[9] Oded Ghitza. A measure of in-synchrony regions in the auditory nerve firing patterns as a basis for speech vocoding. In *Proceedings ICASSP-1985*, volume 2, pages 505–508, 1985.

[10] B.A. Hanson and D.Y. Wong. The harmonic spectral suppression (hms) technique for intelligibility enhancement in the presence of interfering speech. In *Proceedings ICASSP-1984*, pages 18A.5.1–18A.5.4, 1984.

[11] Luc M. Van Immerseel and Jean-Pierre Martens. Pitch and voiced/unvoiced determination with an auditory model. *Journal of the Acoustical Society of America*, 91(6):3511–3526, June 1992.

[12] R.W. Johnson and J.E. Shore. Minimum cross-entropy spectral analysis of multiple signals. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP–31:574–582, 1983.

[13] S. Lawrence Marple Jr. *Digital Spectral Analysis with Applications.* Prentice-Hall, 1987.

[14] Gary E. Kopec and Marcia A. Bush. An lpc-based spectral similarity measure for speech recognition in the presence of co-channel speech interference. In *Proceedings ICASSP-1989*, pages 270–273, 1989.

[15] J. Kupin. Wire — a wireline simulator. CCR-P, April 1993. [software].

[16] C.K. Lee and D.G. Childers. Cochannel speech separation. *Journal of the Acoustical Society of America*, 83(1):274–280, January 1988.

[17] J. C. R. Licklider. A duplex theory of pitch perception. *Experientia*, 7:128–133, 1951.

[18] R. F. Lyon. Experiments with a computational model of the cochlea. In *Proceedings ICASSP-1986*, volume 3, pages 1975–1978, 1986.

[19] C. M. H. Marin and S. McAdams. Segregation of concurrent sounds. ii: Effects of spectral envelope tracing, frequency modulation coherence, and frequency modulation width. *Journal of the Acoustical Society of America*, 89(1):341–351, January 1990.

[20] J. S. Marques and L. B. Almeida. Frequency-varying sinusoidal modeling of speech. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 37(5):763–765, 1989.

[21] S. McAdams. Segregation of concurrent sounds. i: Effects of frequency modulation coherence. *Journal of the Acoustical Society of America*, 86(6):2148–2159, December 1989.

[22] Y. Medan, E. Yair, and D. Chazan. Super resolution pitch determination of speech signals. *IEEE Transactions on Signal Processing*, 39(1):40–48, 1991.

[23] Ray Meddis and Michael J. Hewitt. Virtual pitch and phase sensitivity of a computer model of the auditory periphery. i: Pitch identification. *Journal of the Acoustical Society of America*, 89(6):2866–2882, June 1991.

[24] Ray Meddis and Michael J. Hewitt. Virtual pitch and phase sensitivity of a computer model of the auditory periphery. ii: Phase sensitivity. *Journal of the Acoustical Society of America*, 89(6):2883–2894, June 1991.

[25] K. Min, D. Chien, S. Li, and C. Jones. Automated two speaker separation system. In *Proceedings ICASSP-1988*, pages 537–540, 1988.

[26] J. Naylor and J. Porter. An effective speech separation system which requires no a priori information. In *Proceedings ICASSP-1991*, pages 937–940, 1991.

[27] J.A. Naylor and S.F. Boll. Techniques for suppression of an interfering talker in co-channel speech. In *Proceedings ICASSP-1987*, pages 6.12.1–6.12.4, 1987.

[28] Takami Niihara and Seiji Inokuchi. Transcription of sung song. In *Proceedings ICASSP-1986*, pages 1277–1280, 1986.

[29] A. M. Noll. Cepstrum pitch determination. *Journal of the Acoustical Society of America*, 41:293–309, February 1967.

[30] Thomas W. Parsons. Separation of speech from interfering speech by means of harmonic selection. *Journal of the Acoustical Society of America*, 60(4):911–918, 1976.

[31] T.F. Quatieri and R.G.Danisewicz. An approach for co-channel talker interfernce suppression using a sinusoidal model for speech. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 38(1):56–69, 1990.

[32] Lawrence R. Rabiner and Ronald W. Schafer. *Digital Processing of Speech Signals.* Prentice-Hall, 1978.

[33] M. Savic, H. Gao, and J. S. Sorenson. Co-channel speaker separation based on maximum-likelihood deconvolution. In *Proceedings ICASSP-1994*, pages I–25–I–28, 1994.

[34] M. R. Schroeder. Period histogram and product spectrum: new methods for fundamental frequency measurements. *Journal of the Acoustical Society of America*, 43:829–834, 1968.

[35] Stephanie Seneff. A joint synchrony/mean-rate model of auditory speech processing. *Journal of Phonetics*, 16:55–76, 1988.

[36] S. A. Shamma. Speech processing in the auditory system i: the representation of speech sounds in the responses of the auditory nerve. *Journal of the Acoustical Society of America*, 78(5):1612–1621, 1985.

[37] Malcolm Slaney. Auditory toolbox: A matlab toolbox for auditory modeling work. Technical Report 45, Apple Computer Inc., Advanced Technology Group, 1994.

[38] Malcolm Slaney and Richard F. Lyon. A perceptual pitch detector. In *Proceedings ICASSP-1990*, pages 357–360, 1990.

[39] Malcolm Slaney, Daniel Naar, and Richard F. Lyon. Auditory model inversion for sound separation. In *Proceedings ICASSP-1994*, pages II–77–II–80, 1994.

[40] Y. Stettiner, D. Malah, and D. Chazan. Estimation of the parameters of a long-term model for accurate representation of voiced speech. In *Proceedings ICASSP-1993*, pages II–534–II–537, 1993.

[41] R.J. Webster. Spectral line profiles generated by deterministic frequency modulation. *IEEE Transactions on Signal Processing*, 39(4):1012–1017, 1991.

[42] Mitchel Weintraub. The grasp sound separation system. In *Proceedings ICASSP-1984*, pages 18A.6.1–18A.6.4, 1984.

[43] Mitchel Weintraub. A computational model for separating two simultaneous talkers. In *Proceedings ICASSP-1986*, pages 3.1.1–3.1.4, 1986.

[44] J.D. Wise, J.A. Caprio, and T.W. Parks. Maximum likelihood pitch estimation. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP–24(5):418–423, October 1976.

MISSION

OF

ROME LABORATORY

Mission. The mission of Rome Laboratory is to advance the science and technologies of command, control, communications and intelligence and to transition them into systems to meet customer needs. To achieve this, Rome Lab:

    a. Conducts vigorous research, development and test programs in all applicable technologies;

    b. Transitions technology to current and future systems to improve operational capability, readiness, and supportability;

    c. Provides a full range of technical support to Air Force Materiel Command product centers and other Air Force organizations;

    d. Promotes transfer of technology to the private sector;

    e. Maintains leading edge technological expertise in the areas of surveillance, communications, command and control, intelligence, reliability science, electro-magnetic technology, photonics, signal processing, and computational science.

The thrust areas of technical competence include: Surveillance, Communications, Command and Control, Intelligence, Signal Processing, Computer Science and Technology, Electromagnetic Technology, Photonics and Reliability Sciences.